
1 Octave for Chapter 9

Octave for Chapter 9

Examples and Templates in this section provide sample Octave code for least squares, and QR -factorization. You can access our code through the link at the bottom of each template. Feel free to modify the code and experiment to learn more!

You can write your own code using Octave software or online Octave cells. To access Octave cells online, go to the [Sage Math Cell Webpage](#), select OCTAVE as the language, enter your code, and press EVALUATE.

To "save" or share your online code, click on the *Share* button, select *Permalink*, then copy the address directly from the browser window. You can store this link to access your work later or share this link with others. You will need to get a new Permalink every time you modify the code.

Octave Tutorial

Least Squares

The following theorem establishes one way to implement least squares.

Theorem 1 (??). Let A be an $m \times n$ matrix, let $\mathbf{b}$ be a column vector in $\mathbb{R}^m$. Consider the matrix equation

$$A\mathbf{x} = \mathbf{b}$$

1. Any solution $\mathbf{z}$ to the normal equations

$$(A^T A)\mathbf{z} = A^T \mathbf{b}$$

is a best approximation to a solution to $A\mathbf{x} = \mathbf{b}$ in the sense that $\|\mathbf{b} - A\mathbf{z}\|$ is minimized.

2. If the columns of A are linearly independent, then $A^T A$ is invertible and $\mathbf{z}$ is given uniquely by

$$\mathbf{z} = (A^T A)^{-1} A^T \mathbf{b}$$

We can implement this process as follows.

Template 1. We show how to find a least squares solution to $A\mathbf{x} = \mathbf{b}$.

```
% Define matrix A
A=[9 -3 3;
  1 -1 1;
  4 0 1;
  1 1 1;
  9 -1 2];

% Define vector b
```

```
b=[3;1;1;2;4];
```

```
% Now we solve for z
z=inv(transpose(A)*A)*transpose(A)*b
```

[Link to code.](#)

Warning 1. Finding matrix inverses is both computationally expensive and unstable.

To do least squares in Octave, we use the backslash operator (`\`). Depending on the type of matrix, backslash uses different techniques to compute the least squares solution more efficiently. For more information about *inv* and the backslash operator (`\`) see [Reference](#).

The following two examples illustrate the two implementations of least squares side-by-side.

Example 1. We will compare the results of the two implementations of least squares for a 7×3 coefficient matrix.

```
% Define matrix A
A=[4 -1 0;
   2 -1 1;
   6 3 1;
  -1 1 2;
   3 1 -2;
  -2 8 7;
   1 1 -5];

% Define vector b
b=[-2;4;1;2;-5;1;-3];

% Implement least squares using our original method
z1=inv(transpose(A)*A)*transpose(A)*b

% Implement least squares using the backslash operator
z2=A\b
```

[Link to code.](#)

Example 2. We will compare the performance of two the implementations of least squares for a 2000×20 coefficient matrix.

```
% Define matrix A
A=rand(2000,20); % rand(m,n) creates an m by n matrix whose entries are between 0 and 1

% Define vector b
b=rand(2000,1);

% Start the timer
```

```
% using rref
A_b=[A b];
rref(A_b)
```

[Link to code.](#)

Warning 2. While the backslash (\) operator can be used to solve feasible systems, we urge you to be very cautious because

- If a system has infinitely many solutions, the backslash (\) operator will find one particular solution.
- If a system has no solutions, the backslash (\) operator will find an approximate solution using least-squares.

The way the answer is presented, you cannot tell the difference!

QR-factorization

Recall the definition of *QR*-factorization.

Definition 1 (??). Let A be an $m \times n$ matrix with independent columns. A *QR-factorization* of A expresses it as $A = QR$ where Q is $m \times n$ with orthonormal columns and R is an invertible and upper triangular matrix with positive diagonal entries.

Example 4. Find the *QR*-factorization of

$$A = \begin{bmatrix} 2 & -1 & 4 & 0 \\ 3 & 1 & 1 & -2 \\ 1 & 5 & -1 & 6 \\ -3 & 4 & 0 & 7 \end{bmatrix}$$

and demonstrate that matrix Q is orthogonal (columns are orthonormal).

Explanation. % Define A

```
A=[2 -1 4 0;
   3 1 1 -2;
   1 5 -1 6;
  -3 4 0 7];

% Find the QR-factorization of A
[Q,R]=qr(A)

% Verify that columns of Q are orthonormal
% You may get numbers that are close to zero but not zero
% due to round-off error
transpose(Q)*Q

% why does this verify that the columns are orthonormal?
```

[Link to code.](#)

Question 1 In the code above, we used the product $Q^T Q$ to verify that the columns of Q are orthonormal. What do you expect this product to look like if the columns are orthonormal? What will the product look like if they are not? You will be asked to explain your reasoning in Problem ??.

Recall the following algorithm for using QR -factorization to approximate eigenvalues from [QR-Factorization](#).

Algorithm 1 (??). Let A be an invertible matrix.

Step 1: Define $A_1 = A$ and factor it as $A_1 = Q_1 R_1$.

Step 2: Define $A_2 = R_1 Q_1$ and factor it as $A_2 = Q_2 R_2$.

Step 3: Define $A_3 = R_2 Q_2$ and factor it as $A_3 = Q_3 R_3$.

$\vdots$

Step $k + 1$: Define $A_{k+1} = R_k Q_k$ (where $A_k = Q_k R_k$).

Note that A_{k+1} is similar to A_k (in fact, $A_{k+1} = R_k Q_k = (Q_k^{-1} A_k) Q_k$), and hence each A_k has the same eigenvalues as A . If the eigenvalues of A are real and have distinct absolute values, the sequence of matrices $A_1, A_2, A_3, \dots$ converges to an upper triangular matrix with these eigenvalues on the main diagonal.

Compare the following example to Example ??.

Example 5. Use the QR algorithm to approximate the eigenvalues of $A = \begin{bmatrix} 1 & 1 \\ 2 & 0 \end{bmatrix}$.

Explanation. Here is a very simple implementation of the algorithm.

```
% Define A
A=[1 1;
   2 0];

% Set the desired number of iterations
iter=20;

% Implement the algorithm. Note that we are over-writing the previous A with every iteration.
for i=1:iter
    [Q,R]=qr(A);
    A=R*Q
end
```

[Link to code.](#)

Observe that the main diagonal entries are approaching the eigenvalues of A .

Octave Exercises

Problem 2 Attempt to solve each of the following systems of equations using the backslash ($\backslash$) operator, and the `rref` function. Interpret your results.

1.

$$\begin{array}{rclcrcl} 3x & - & y & + & 4z & = & 2 \\ & & y & + & z & = & 1 \\ -2x & & & + & 3z & = & 1 \end{array} \quad (2)$$

What is true about this system?

Multiple Choice:

- (i) The system is consistent. Using the backslash operator produces the same result as using the rref function.
- (ii) The system is inconsistent. The backslash operator gives the least squares approximation.
- (iii) The system is consistent but has infinitely many solutions. The backslash operator gives one particular solution.

2.

$$\begin{array}{rclcrcl} 3x & - & y & + & 4z & = & 2 \\ x & - & y & + & 2z & = & 4 \\ -2x & + & y & - & 3z & = & -3 \end{array} \quad (3)$$

What is true about this system?

Multiple Choice:

- (i) The system is consistent. Using the backslash operator produces the same result as using the rref function.
- (ii) The system is inconsistent. The backslash operator gives the least squares approximation.
- (iii) The system is consistent but has infinitely many solutions. The backslash operator gives one particular solution.

3.

$$\begin{array}{rclcrcl} 3x & - & y & + & 4z & = & 2 \\ x & - & y & + & 2z & = & 1 \\ -2x & + & y & - & 3z & = & 1 \end{array} \quad (4)$$

What is true about this system?

Multiple Choice:

- (i) The system is consistent. Using the backslash operator produces the same result as using the rref function.
- (ii) The system is inconsistent. The backslash operator gives the least squares approximation.
- (iii) The system is consistent but has infinitely many solutions. The backslash operator gives one particular solution.

Problem 3 Use least squares to find a line of best fit for the points shown in the GeoGebra interactive below. Enter the equation of the line into the GeoGebra interactive to view your result.

GeoGebra link: <https://www.geogebra.org/m/kmt5ukhc>

1. Find the residual, $\|\mathbf{b} - \mathbf{Az}\|$, for your line of best fit.
2. The original line, $y = x + 1$ initially shown in the interactive, visually appeared to be a pretty good fit. Find the residual for the line $y = x + 1$ and compare it to the residual you got for your line. Is your line a better fit?

Problem 4 Explain how to interpret the product $Q^T Q$ to determine whether matrix Q has orthonormal columns. Prove your claims.

Problem 5 Modify the code in Example ?? to approximate the eigenvalues of $A = \begin{bmatrix} 9 & 2 & 8 \\ 2 & -6 & -2 \\ -8 & 2 & -5 \end{bmatrix}$. Compare your answers to the answers you got for Problem ??.

Problem 6 Modify the code in Example ?? for the matrix $A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. What are you observing? Are we getting close to finding the eigenvalues of A ? Explain what is happening and why.