

MATLAB Project 2: MATH 461, Spring 2021

This page is more information which can be helpful for your MATLAB work, including some new commands. You're responsible for knowing what's been done before. The project starts on the next page. The previous guidelines on format still hold.

1. New Commands

- (a) Powers of matrices can be done in the obvious way. If A is a matrix then we can do:
- ```
>> A^2
```
- or:
- ```
>> A^17
```
- (b) Matrix multiplication can be done with:
- ```
>> A*B
```
- This also takes care of matrices times vectors. For example if
- ```
>> A=[1 2 3;4 5 6;7 8 9]
```
- ```
>> v=[-2 ; 1 ; 3]
```
- then we can do the product  $Av$  with
- ```
>> A*v
```
- (c) The determinant of a matrix is done with the `det` command. This can be done either on a given matrix:
- ```
>> A=[1 2 3;4 5 6;7 8 9]
```
- ```
>> det(A)
```
- or inline:
- ```
>> det([1 2 3;4 5 6;7 8 9])
```
- (d) The command `eye(n)` will produce the  $n \times n$  identity matrix  $I_n$ . If  $A$  is a  $3 \times 3$  matrix that we have already entered, then we can type
- ```
>> M = [A eye(3)]
```
- to produce the 3×6 matrix $[A \ I_3]$ as in the algorithm for computing A^{-1} . (Aside: if \mathbf{b} is a column vector, you can also create augmented matrices by typing $[A \ \mathbf{b}]$) We could then do $N = \text{rref}(M)$, followed by
- ```
>> X = N(:,4:6)
```
- to extract the inverse from  $N$ . This command is extracting the submatrix which consists of all rows, but only columns 4 through 6. However, MATLAB has a command to compute an inverse all at once. The inverse of  $A$  can be done with the `inv` command:
- ```
>> inv(A)
```
- or inline as with `det`.

2. Combining Functions

MATLAB allows you to combine commands easily. For example, suppose we wish to solve the system

$$\begin{aligned} 2x_1 + x_2 &= 0.5 \\ -3x_1 - 5x_2 &= 10 \end{aligned}$$

We can do this with a single MATLAB command line entry with an inverse matrix:

```
>> inv([2 1;-3 -5])*[0.5;10]
```

MATH 461 Spring 2021 MATLAB Project 2 – due Mon, 3/8 by 11:59 pm

- **Directions:** The same directions from the previous MATLAB project apply here. Please reread them. You will need to write a script (.m file) and then “publish” it in MATLAB as a pdf in order to turn it in on Gradescope.
- **Format:** Each problem specifies whether to use `format rat` (fractions) or `format short` (decimals). Format `rat` is good if you are actually working with rational numbers. But if you have irrational numbers, it will give rational approximations for them. For example, in `format rat`, $\pi = 355/113$ and $\sqrt{2} = 1393/985$. Approximations like this can be a bit misleading, so it is sometimes better to work with decimals.

On to the project problems:

1. (Use `format short`, except when specified otherwise) Recall that a counter-clockwise rotation about the origin by angle θ defines a linear transformation from \mathbb{R}^2 to \mathbb{R}^2 . As discussed in class, the standard matrix of this transformation is

$$R_\theta = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}.$$

- (a) Enter $A = R_\theta$ for $\theta = \pi/10$ in MATLAB. (Note: you can type things like `cos(pi/5)` in MATLAB.) Then use it to rotate the vector $\mathbf{v} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ counter-clockwise by an angle of $\pi/10$.
- (b) Let $B = R_\theta$ for $\theta = \pi/13$. Use MATLAB to determine if $AB = BA$.
- (c) \star What does the previous result say about how these two different rotation operations interact with each other?
- (d) The composition of two rotations is also a rotation. Let $C = AB$, and we shall determine the angle of rotation for the composition. We know this matrix has the form R_θ for some θ . One way to determine θ is to extract the (1,1) entry and take its arccosine (this assumes θ is between 0 and π). This can be accomplished in MATLAB by typing `t = acos(C(1,1))`. In this case, `t` is a rational multiple of π . Determine `t` exactly by switching to `format rat` and computing `t/pi`.
- (e) Switch back to `format short`. The inverse of R_θ is $R_{-\theta}$. Verify this in MATLAB for $\theta = \pi/10$ by comparing `inv(A)` with $R_{-\pi/10}$.
- (f) Reflection about a line through the origin in \mathbb{R}^2 is also a linear transformation. Let L_θ denote the standard matrix for the reflection about the line through the origin which makes an angle of θ with the positive x_1 -axis. For example, L_0 is just reflection about the x_1 -axis, and is given by

$$L_0 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

It can be shown that the matrix L_θ is given by

$$L_\theta = R_\theta L_0 R_{-\theta}.$$

(To understand this, you must convince yourself that the triple composition on the right performs the desired reflection. Think about what that composition does to a vector in the direction of the line we are reflecting about, and also what it does to a vector perpendicular to this line.) Compute the matrix L_θ for $\theta = \pi/10$.

- (g) Determine if $L_{\pi/10}L_0 = L_0L_{\pi/10}$.
- (h) It can be shown that the composition of two reflections is a rotation. Determine the angle of rotation of $L_{\pi/10}L_0$. (As before, present your answer as a rational multiple of π .)
2. (Use `format rat`) Let $A = \begin{bmatrix} 3 & 2 & 1 \\ 7 & 2 & 4 \\ 7 & 2 & 6 \end{bmatrix}$.

- (a) Follow the instructions in the introduction on how to enter the augmented matrix $[A \ I_3]$, put it in reduced row echelon form, and then extract the inverse matrix A^{-1} .
- (b) Now let MATLAB compute the inverse of A directly using the `inv` command. Make sure your answer coincides with the previous part.
3. (a) (Use `format rat`) Use MATLAB to compute the determinants of the following two matrices.

$$A = \begin{bmatrix} -2 & 0 & 0 & 0 \\ 16 & 2 & 0 & 0 \\ 3 & -7 & -1 & 0 \\ 9 & 3 & 4 & 4 \end{bmatrix} \quad B = \begin{bmatrix} 2 & 0 & 1 & -1 \\ 1 & 3 & 2 & 3 \\ 0 & 2 & 3 & 2 \\ 3 & 3 & 1 & 0 \end{bmatrix}$$

- (b) ★ The determinant of A could have been computed easily without MATLAB. What general fact could have been used to do this?
- (c) Use MATLAB to compute the matrix $C = AB$ and also $\det(C)$.
- (d) ★ The determinant of C could have been determined without having MATLAB compute $\det(C)$. What general fact could have been used to do this?
4. (Use `format rat`) Let $A = \begin{bmatrix} 8 & 2 & 0 & 6 \\ 1 & 0 & 5 & 4 \\ 5 & -1 & 2 & 0 \\ 5 & 3 & 7 & 8 \end{bmatrix}$.

- (a) Compute $\det A$.
- (b) ★ Consider the following matrices obtained from A by elementary row operations:
- B is obtained from A by swapping row 2 with row 3.
 - C is obtained from A by multiplying row 3 by -2 .
 - D is obtained from A by subtracting row 4 from row 1.
- Without doing any computations in MATLAB, determine the values of $\det B$, $\det C$, and $\det D$.
- (c) Use MATLAB to produce the matrices B, C, D by applying the row operations to A (see Project 1 for how to do row operations in MATLAB.) Since you don't want to actually change what you've stored as A in MATLAB, you may want to start with the command `B = A`, and then do the row operation to B (and similarly for C and D).
- (d) Compute the determinants of B, C , and D in MATLAB, and verify that they are the same as the values you predicted above.

5. (a) Symbolically define the variables a through d , using the command `syms a b c d`. Then enter the matrix $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$.
- (b) Use the `inv` command to compute A^{-1} . This is what the formula for the inverse of a 2×2 matrix looks like. Note that every entry has a denominator of $\det A = ad - bc$.
- (c) Define more symbolic variables and the matrix $B = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$. As above, use the `inv` command to compute B^{-1} . This is the formula for the inverse of a 3×3 matrix.
- (d) The formula is a bit of a mess because every entry has a denominator of $\det B$, just as in the 2×2 case. Multiply B^{-1} by the scalar $\det B$ to clear out those denominators and simplify what you are looking at.
- (Note: the inverse of an $n \times n$ matrix A can always be written as $A^{-1} = \frac{1}{\det A} \text{adj } A$, where $\text{adj } A$ is an $n \times n$ matrix called the *adjugate* of A . The entries of $\text{adj } A$ are cofactors of A . This formula is a consequence of *Cramer's Rule*, see §3.3 if you are interested in the details. In part (d) of this problem, you computed $\text{adj } B$.)