Lecturer: Daoud Siniora
MACT 2132

Linear Algebra

Course project
Summer 2025

AUC

The Inverse and Adjoint of a Matrix

Submission

- You are encouraged to work in groups of at most 3 students.
- You need to run your code with the TA.
- Plagiarizing your project will result in getting zero credit and will trigger an academic integrity case. Your submissions will be uploaded to Turnitin.
- You are encouraged to start working on your projects as soon as you can, since it might take some time and effort on your part.

Mission 1: A Programming code.

Design a code using your preferred programming language: C++, Python, MATLAB, etc, to do the following tasks.

- Input: A 4×4 matrix A.
- Output 1: An RREF matrix row equivalent to A.
- Output 2: The determinant of A.
- Output 3: The inverse of A, or output a message in case the matrix was not invertible.

To perform the tasks above you need to make use of the following from lecture notes.

- Row-Echelon Form Algorithm, see Algorithm 1.4.3 page 35.
- Matrix Inverse Algorithm, see Algorithm 2.1.6 page 48.
- Section 2.4 (Determinant and EROs) page 65.

Mission 2: A LaTeX paper.

You are required to Submit a well-written, organized paper using LaTeX (overleaf). The Latex paper consists of the three sections explained below that describe your project. The project revolves around finding the inverse of a matrix via two methods:
First, using Gauss-Jordan as done in the algorithm in Mission 1.
Second, using the adjoint of a matrix.

Section 1: Introduction and Preliminaries

Here you motivate your project and for the purpose of making your paper somewhat self-contained, you are required to include definitions and theorems covered in the course that you are going to use in your proofs of Section 3. You may also add some history of the subject matter.

Section 2: Code

Include the code you developed in Mission 1. Moreover, use your algorithm to search for and provide two *interesting* examples of 4×4 matrices: an invertible matrix $\mathbf{M}$ and a noninvertible matrix $\mathbf{N}$. Include screen shots of the output of your algorithm for matrices M and N.

Section 3: Adjoint of a Matrix

Let A be any square matrix. The *cofactor matrix* of A, denoted by $\operatorname{cof}(A)$, is the matrix whose (i, j)-entry is the (i, j)-cofactor C_{ij} of the matrix A. The *adjoint* of A, denoted by $\operatorname{adj}(A)$, is defined to be the transpose of its cofactor matrix.

$$\operatorname{adj}(A) = (\operatorname{cof}(A))^T$$

The adjoint is also known as *adjugate* or *adjunct*. In this section, you need to do the following.

1. Define what an adjoint of a matrix is.
2. Prove the following theorem. For any $n \times n$ matrix A we have that
$$A \operatorname{adj}(A) = \det(A)\, I_n\,.$$
3. Verify the theorem on the matrices $\mathbf{M}$ and $\mathbf{N}$ you found in Section 2.
4. If a matrix A is invertible, use the theorem to give a formula that computes A^{-1} using its adjoint.
5. Use your formula to compute the inverse of $\mathbf{M}$ and see if this agrees with the output of your algorithm.