Chem237: Lecture 13

Shane Flynn

5/3/18

1 Note

9-9-19; Just copied the lecture notes I have over, this has not been reviewed/edited/made ready for the world.

Gaussian Elimination

Reduce your matrix to an upper-triangular (or lower triangular) form by performing operations such that the determinant doesn't change.

Consider the square N by N matrix **A**, which has column vectors labeled $\vec{a}_i$

$$\mathbf{A} = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{1N} & \cdots & A_{NN} \end{bmatrix} = \begin{bmatrix} \vec{a}_1 & \ldots & \vec{a}_N \end{bmatrix} \tag{1}$$

To compute the determinant of **A** we know the following

$$\det(\mathbf{A}) = \det(\vec{a}_1 \cdots \vec{a}_N) = \det(\vec{a}_1 - \lambda_{a2}, \vec{a}_2 \cdots \vec{a}_N) \tag{2}$$

Let $\lambda_1 = \frac{A_{N1}}{A_{N2}}$ then we have

$$\begin{bmatrix} \star & A_{12} & \cdots & A_{1N} \\ \star & A_{22} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & A_{N2} & \cdots & A_{NN} \end{bmatrix} \tag{3}$$

Next step set $\lambda_2 = \frac{A_{N2}}{A_{N3}}$ then we have

$$\begin{bmatrix} \star & \star & \cdots & \vdots \\ \star & \vdots & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{NN} \end{bmatrix} \tag{4}$$

In this way we can reduce every element of the Nth row to 0 except for the final term A_{NN}. We can then repeat this process for the N-1 row (Second to last row), which will convert every element up to N-1 in the row to 0.

$$\begin{bmatrix} \star & \star & \cdots & \vdots \\ \star & \vdots & \cdots & \vdots \\ 0 & \cdots & 0 & \star \\ 0 & \cdots & 0 & A_{NN} \end{bmatrix} \tag{5}$$

Repeating the process iteratively eventually ends in

$$\begin{bmatrix} \star & \star & \star & \star \\ 0 & \star & \star & \star \\ 0 & 0 & \star & \star \\ 0 & 0 & 0 & \star \end{bmatrix} \tag{6}$$

What happens if you have a 0 at some point during the iterations? Just move it and keep going.

1.1 Numerical Linear Algebra

Using the definition of the determinant the scaling would be N! which is very bad. Using the algorithm from above leads to N^3 scaling, which is a common result for numerical linear algebra algorithms. a_1 - λ_{a2} = 2 operations scaling as N, Appy to row = N, apply to other ros = N.

1.2 Linear Indepedence

Consider a set of n vectors $\vec{a}_1 \cdots \vec{a}_N$. These vectors are linearly indepedent if for any $\lambda_1 \cdots \lambda_N$

$$\sum_{i=1}^{N} \lambda_i \vec{a}_i \neq 0 \tag{7}$$

and vice versa linearly depedent if

$$\sum_{i=1}^{N} \lambda_i \vec{a}_i = 0 \tag{8}$$

Theorem: determinate matrix formed by N vectors $\det(\vec{a}_1 \cdots \vec{1}_N) = 0$ iff $\vec{a}_1 \cdots \vec{a}_N$ are linearly depedent, and vice versa linearly indepedent vectors produce a determinate $\neq 0$.