Gaussian elimination w/o pivoting (LU-factorization)

linalg3

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}$$

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -a_{21}/a_{11} & 1 & 0 & 0 \\ -a_{31}/a_{11} & 0 & 1 & 0 \\ -a_{41}/a_{11} & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -v_2 & 1 & 0 & 0 \\ -v_3 & 0 & 1 & 0 \\ -v_4 & 0 & 0 & 1 \end{pmatrix}$$

M_1^{-1} easy to get.

$$M_1 A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a'_{22} & a'_{23} & a'_{24} \\ 0 & a'_{32} & a'_{33} & a'_{34} \\ 0 & a'_{42} & a'_{43} & a'_{44} \end{pmatrix}$$

$$M_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -a'_{32}/a'_{22} & 1 & 0 \\ 0 & -a'_{42}/a'_{22} & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -v'_3 & 1 & 0 \\ 0 & -v'_4 & 0 & 1 \end{pmatrix}$$

M_2^{-1} easy to compute

$$M_2 M_1 A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a'_{22} & a'_{23} & a'_{24} \\ 0 & 0 & a''_{33} & a''_{34} \\ 0 & 0 & a''_{43} & a''_{44} \end{pmatrix}$$

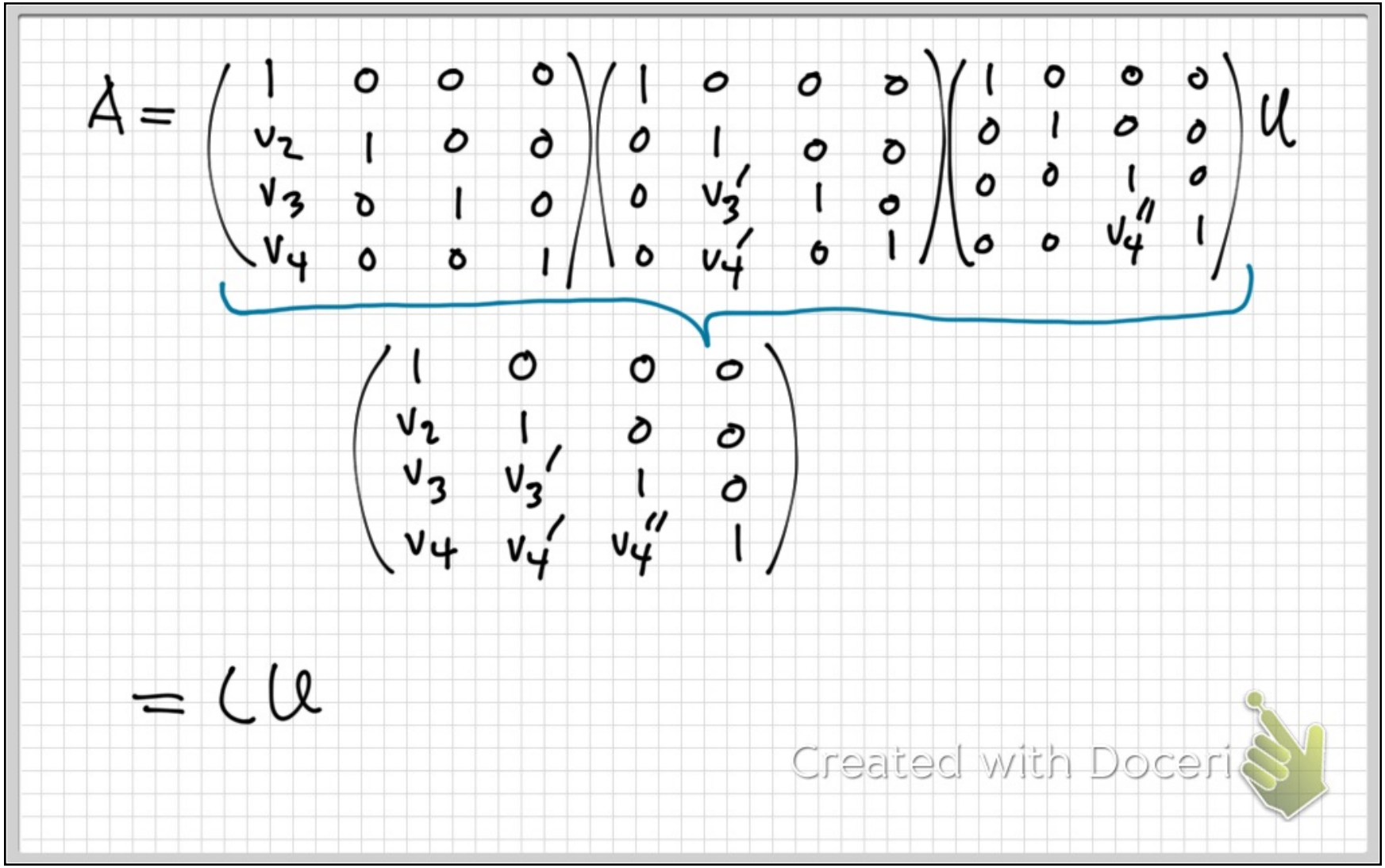
$$M_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -a''_{43}/a''_{33} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -v''_4 & 1 \end{pmatrix}$$

M_3^{-1} easy to compute

$$M_3 M_2 M_1 A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ 0 & a_{22}' & a_{23}' & a_{24}' \\ 0 & 0 & a_{33}'' & a_{34}'' \\ 0 & 0 & 0 & a_{44}''' \end{pmatrix} = U$$

$$\text{so } A = M_1^{-1} M_2^{-1} M_3^{-1} U$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ v_2 & 1 & 0 & 0 \\ v_3 & 0 & 1 & 0 \\ v_4 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & v_3' & 1 & 0 \\ 0 & v_4' & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & v_4'' & 1 \end{pmatrix} U$$


A =
U
= LU
Created with Doceri

Algorithm

```
for k = 1 to n-1
    for j = k+1 to n
        a_jk = a_jk / a_kk
    end
    for j = k+1 to n
        v = a_jk
        for p = k+1 to n
            a_jp = a_jp - v a_kp
        end
    end
end
```

$L = I + \text{tril}(A,-1)$, $U = \text{triu}(A)$

input A

output (L\U)

main line of computational work within 3 loops

work $\sim \frac{2}{3}n^3$

$O(n^3)$ complexity