LECTURE 4: LINEAR ALGEBRA

- We wish to solve for a linear system of equations.
  For now assume equal number of equations as variables *N*.

$$\begin{matrix} a_{11} & x_1 & + & a_{12}\, x_2 & + & \cdots & a_{1m}\, x_n = d_1 \\ a_{21} & x_1 & + & a_{22}\, x_2 & + & \cdots & a_{2m}\, x_n = d_2 \\ \vdots & & & \vdots & & & \vdots \quad\quad \vdots \\ a_{n1} & x_1 & + & a_{n2}\, x_2 & + & \cdots & a_{nm}\, x_n = d_n \end{matrix} \longrightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$$

$Ax = b$: formal solution is $x= A^{-1}b$; A is NxN matrix.
Matrix inversion is very slow.

Gaussian Elimination

- $Ax = b$
- Multiply any row of A by any constant, and do the same on b
- Take linear combination of two rows, adding or subtracting them, and the same on b
- We can keep performing these operations until we set all elements of first column to 0 except 1st one, which can be set to 1
- Then we can repeat the same to set to 0 all elements of 2nd column, except first 2…
- We end up with an upper diagonal matrix with 1 on the diagonal

Gaussian Elimination

- $Ax = b$
- Multiply any row of A by any constant, and do the same on b
- Take linear combination of two rows, adding or subtracting them, and the same on b
- We can keep performing these operations until we set all elements of first column to 0 except 1st one, which can be set to 1
- Then we can repeat the same to set to 0 all elements of 2nd column, except first 2…
- We end up with an upper diagonal matrix with 1 on the diagonal

Gaussian Elimination

$$\begin{pmatrix} 2 & 1 & 4 & 1 \\ 3 & 4 & -1 & -1 \\ 1 & -4 & -1 & 5 \\ 2 & -2 & 1 & 3 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -4 \\ 3 \\ 9 \\ 7 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 3 & 4 & -1 & -1 \\ 1 & -4 & -1 & 5 \\ 2 & -2 & 1 & 3 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 3 \\ 9 \\ 7 \end{pmatrix}$$

$$\downarrow$$

$$\begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 0 & 2.5 & -7 & -2.5 \\ 0 & -4.5 & 1 & 4.5 \\ 0 & -3 & -3 & 2 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 9 \\ 11 \\ 11 \end{pmatrix} \longleftarrow \begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 0 & 2.5 & -7 & -2.5 \\ 1 & -4 & -1 & 5 \\ 2 & -2 & 1 & 3 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 9 \\ 9 \\ 7 \end{pmatrix}$$

$$\downarrow$$

$$\begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 0 & 1 & -2.8 & -1 \\ 0 & -4.5 & 1 & 4.5 \\ 0 & -3 & -3 & 2 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 3.6 \\ 11 \\ 11 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 0 & 1 & -2.8 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 3.6 \\ -2 \\ 1 \end{pmatrix}$$

Backsubstitution

$$\begin{pmatrix} 1 & 0.5 & 2 & 0.5 \\ 0 & 1 & -2.8 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -2 \\ 3.6 \\ -2 \\ 1 \end{pmatrix}$$

$$1 \cdot z = 1 \;\rightarrow\; z = 1$$

$$1 \cdot y + 0 \cdot z = -2 \;\rightarrow\; y = -2$$

$$1 \cdot x + -2.8 \cdot y - 1 \cdot z = 3.6 \;\rightarrow\; x = -1$$

$$1 \cdot w + 0.5 \cdot x + 2 \cdot y + 0.5 \cdot z = -2 \;\rightarrow\; w = 2$$

Back just means we start at the bottom and move up

Pivoting

- What if the first element is 0?
- Swap the rows (partial pivoting) or rows and columns (full pivoting)!
- In practice simply pick the largest element (keep in mind this changes det sign)

$$\begin{pmatrix} 0 & 1 & 4 & 1 \\ 3 & 4 & -1 & -1 \\ 1 & -4 & 1 & 5 \\ 2 & -2 & 1 & 3 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -4 \\ 3 \\ 9 \\ 7 \end{pmatrix}$$

$$\downarrow$$

$$\begin{pmatrix} 3 & 4 & -1 & -1 \\ 0 & 1 & 4 & 1 \\ 1 & -4 & 1 & 5 \\ 2 & -2 & 1 & 3 \end{pmatrix} \begin{pmatrix} w \\ x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 3 \\ -4 \\ 9 \\ 7 \end{pmatrix}$$

LU Decomposition

- We want to solve $Ax = b$ varying b, so we'd like to have a decomposition of A that is done once and then can be applied to several b
- Suppose we can write $\boldsymbol{A = LU}$, where L is lower diagonal and U is upper diagonal: $L(Ux) = b$
- Then we can first solve for y in $Ly = b$ using forward substitution, followed by $Ux = y$ using backward substitution
- Operation count N^3.

Revisit Gauss Elimination: Make A an upper triangular matrix.

- $L_0 A = A'$

$$\underbrace{\frac{1}{a_{00}}\begin{pmatrix} 1 & 0 & 0 & 0 \\ -a_{10} & a_{00} & 0 & 0 \\ -a_{20} & 0 & a_{00} & 0 \\ -a_{30} & 0 & 0 & a_{00} \end{pmatrix}}_{L_0 \text{ lower triangular}} \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} 1 & b_{01} & b_{02} & b_{03} \\ 0 & b_{11} & b_{12} & b_{13} \\ 0 & b_{21} & b_{22} & b_{23} \\ 0 & b_{31} & b_{32} & b_{33} \end{pmatrix}$$

- $L_1 A' = A''$

$$\underbrace{\frac{1}{b_{11}}\begin{pmatrix} b_{11} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -b_{21} & b_{11} & 0 \\ 0 & -b_{31} & 0 & b_{11} \end{pmatrix}}_{L_1 \text{ lower triangular}} \begin{pmatrix} 1 & b_{01} & b_{02} & b_{03} \\ 0 & b_{11} & b_{12} & b_{13} \\ 0 & b_{21} & b_{22} & b_{23} \\ 0 & b_{31} & b_{32} & b_{33} \end{pmatrix} = \begin{pmatrix} 1 & c_{01} & c_{02} & c_{03} \\ 0 & 1 & c_{12} & c_{13} \\ 0 & 0 & c_{22} & c_{23} \\ 0 & 0 & c_{32} & c_{33} \end{pmatrix}$$

Get L_2, L_3 similarly

$$\boxed{\boxed{L_3 L_2 L_1 L_0}\, A}\, x = L_3 L_2 L_1 L_0 b$$

Lower triangular

Upper triangular

Define $L \equiv L_0^{-1} L_1^{-1} L_2^{-1} L_3^{-1}, \quad U \equiv L_3 L_2 L_1 L_0 A$

$$\rightarrow LU = A$$

- $L_3 L_2 L_1 L_0 A$ (= U) is upper diagonal. It gives: $\begin{pmatrix} 1 & \# & \# & \# \\ 0 & 1 & \# & \# \\ 0 & 0 & 1 & \# \\ 0 & 0 & 0 & 1 \end{pmatrix}$
- L is lower diagonal because:

$$L_0 = \frac{1}{a_{00}} \begin{pmatrix} 1 & 0 & 0 & 0 \\ -a_{10} & a_{00} & 0 & 0 \\ -a_{20} & 0 & a_{00} & 0 \\ -a_{30} & 0 & 0 & a_{00} \end{pmatrix} \rightarrow L_0^{-1} = \frac{1}{a_{00}} \begin{pmatrix} a_{00} & 0 & 0 & 0 \\ a_{10} & 1 & 0 & 0 \\ a_{20} & 0 & 1 & 0 \\ a_{30} & 0 & 0 & 1 \end{pmatrix}$$

$$\boxed{\boxed{L_3 L_2 L_1 L_0}\, A}\, x = L_3 L_2 L_1 L_0 b$$

Lower triangular   Upper triangular

Define $L \equiv L_0^{-1} L_1^{-1} L_2^{-1} L_3^{-1}, \quad U \equiv L_3 L_2 L_1 L_0 A$

$\rightarrow LU = A$

- $L_3 L_2 L_1 L_0 A$ (= U) is upper diagonal. It gives:

$$\begin{pmatrix} 1 & \# & \# & \# \\ 0 & 1 & \# & \# \\ 0 & 0 & 1 & \# \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- L is lower diagonal because:

$$L = L_0^{-1} L_1^{-1} L_2^{-1} L_3^{-1} = \frac{1}{a_{00}} \begin{pmatrix} a_{00} & 0 & 0 & 0 \\ a_{10} & b_{11} & 0 & 0 \\ a_{20} & b_{21} & c_{22} & 0 \\ a_{30} & b_{31} & c_{32} & d_{33} \end{pmatrix}$$

We have $N(N+1)/2$ components for L and $N(N-1)/2$ for U because $U_{ii}=1$. So in total N^2 as in A

What if the matrix A is symmetric and positive definite?

- $A_{12}=A_{21}$ hence we can set $U=L^T$, so the LU decomposition is $A=LL^T$
- Symmetric matrix has $N(N+1)/2$ elements, same as L
- This is the fastest way to solve such a matrix and is called Cholesky decomposition (still N^3)
- Pivoting is needed for LU, while Cholesky is stable even without pivoting
- Use numpy.linalg import solve
- L can be viewed as a square root of A, but this is not unique

Inverse and Determinantc

- $AX = I$ and solve with LU (use inv in linalg)
- det $A = L_{00}L_{11}L_{22}$... (note that $U_{ii} = 1$) times number of row permutations
- Better to compute $ln\ detA = lnL_{00} + lnL_{11} + \ldots$

Tridiagonal and Banded Matrices

- Solved with Gaussian substitution: $O(N)$ instead of N^3 in CPU, N instead of N^2 in storage

$$A = \begin{pmatrix} a_{00} & a_{01} & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & 0 & 0 \\ 0 & a_{21} & a_{22} & a_{23} & 0 \\ 0 & 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{43} & a_{44} \end{pmatrix}$$

Tridiagonal

E.g.

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 3 & 4 & -5 & 0 \\ 0 & -4 & 3 & 5 \\ 0 & 0 & 1 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0.5 & 0 & 0 \\ 3 & 4 & -5 & 0 \\ 0 & -4 & 3 & 5 \\ 0 & 0 & 1 & 3 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 0.5 & 0 & 0 \\ 0 & 2.5 & -5 & 0 \\ 0 & -4 & 3 & 5 \\ 0 & 0 & 1 & 3 \end{pmatrix}$$

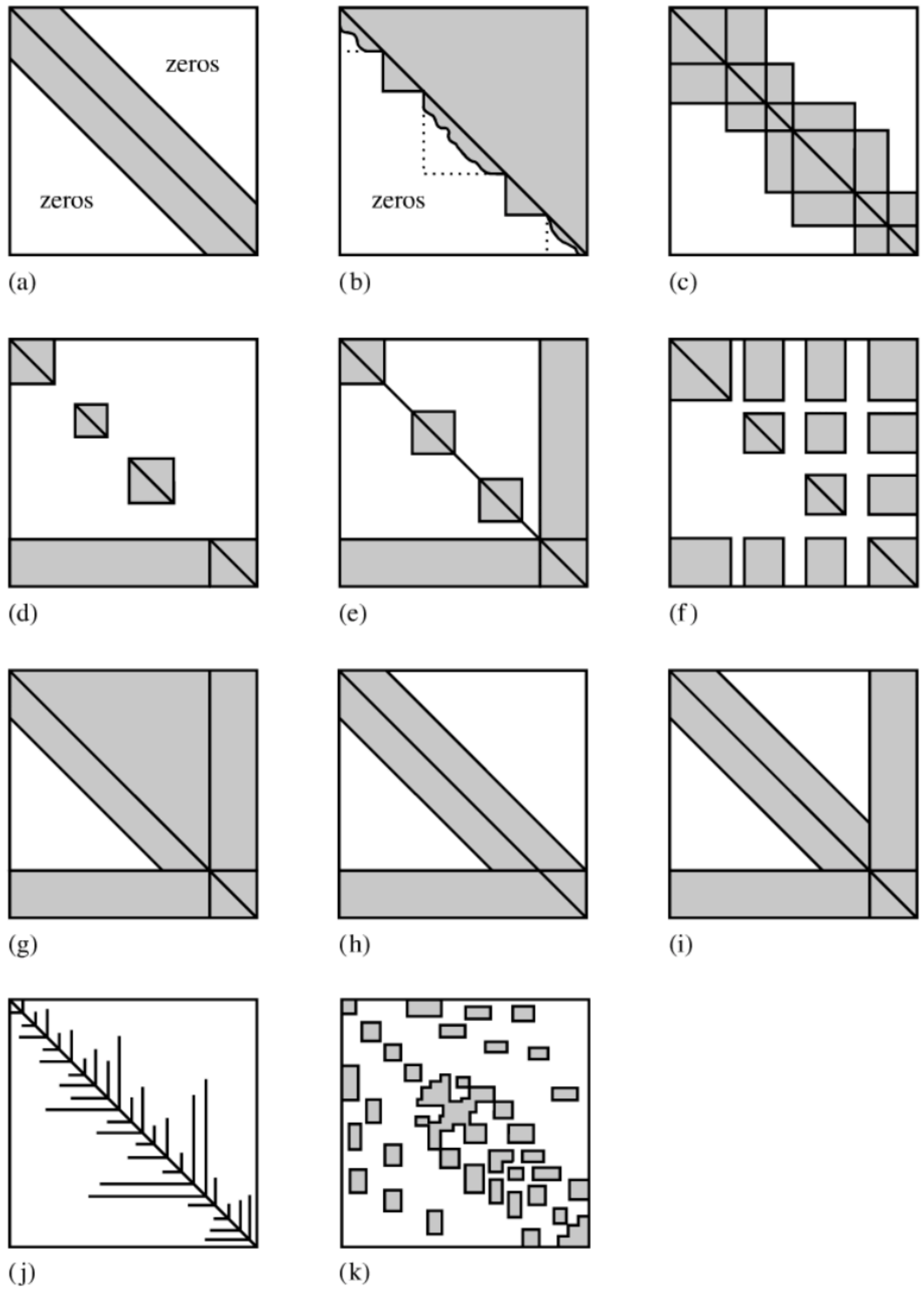
$$\downarrow$$

$$\begin{pmatrix} 1 & 0.5 & 0 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \leftarrow \begin{pmatrix} 1 & 0.5 & 0 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 4 \end{pmatrix} \leftarrow \begin{pmatrix} 1 & 0.5 & 0 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & -5 & 5 \\ 0 & 0 & 1 & 3 \end{pmatrix}$$

Same approach can be used for banded matrices

General sparse matrices

- Allow solutions to scale faster than N^3
- General advice: be aware that such matrices can have much faster solutions
- General advice: be aware that such matrices can have much faster solutions



Credit: NR, Press etal, chapter 2.7

Sherman-Morrison Formulac

- If we have a matrix A we can solve (e.g. tridiagonal etc) and we can add rank 1 component then we can get A^{-1} in $3N^2$:

$$\mathbf{A} \rightarrow (\mathbf{A} + \mathbf{u} \otimes \mathbf{v})$$

$$\begin{aligned}
(\mathbf{A} + \mathbf{u} \otimes \mathbf{v})^{-1} &= (\mathbf{1} + \mathbf{A}^{-1} \cdot \mathbf{u} \otimes \mathbf{v})^{-1} \cdot \mathbf{A}^{-1} \\
&= (\mathbf{1} - \mathbf{A}^{-1} \cdot \mathbf{u} \otimes \mathbf{v} + \mathbf{A}^{-1} \cdot \mathbf{u} \otimes \mathbf{v} \cdot \mathbf{A}^{-1} \cdot \mathbf{u} \otimes \mathbf{v} - \ldots) \cdot \mathbf{A}^{-1} \\
&= \mathbf{A}^{-1} - \mathbf{A}^{-1} \cdot \mathbf{u} \otimes \mathbf{v} \cdot \mathbf{A}^{-1} (1 - \lambda + \lambda^2 - \ldots) \\
&= \mathbf{A}^{-1} - \frac{(\mathbf{A}^{-1} \cdot \mathbf{u}) \otimes (\mathbf{v} \cdot \mathbf{A}^{-1})}{1 + \lambda} \qquad (2.7.2)
\end{aligned}$$

where

$$\lambda \equiv \mathbf{v} \cdot \mathbf{A}^{-1} \cdot \mathbf{u} \qquad (2.7.3)$$

$$\mathbf{z} \equiv \mathbf{A}^{-1} \cdot \mathbf{u} \qquad \mathbf{w} \equiv (\mathbf{A}^{-1})^T \cdot \mathbf{v} \qquad \lambda = \mathbf{v} \cdot \mathbf{z} \qquad (2.7.4)$$

to get the desired change in the inverse

$$\mathbf{A}^{-1} \quad \rightarrow \quad \mathbf{A}^{-1} - \frac{\mathbf{z} \otimes \mathbf{w}}{1 + \lambda} \qquad (2.7.5)$$

Example: cyclic tridiagonal systems

- This happens for finite difference differential equations with periodic boundary conditions

$$\begin{bmatrix} b_0 & c_0 & 0 & \cdots & & & \beta \\ a_1 & b_1 & c_1 & \cdots & & & \\ & & & \cdots & & & \\ & & & \cdots & a_{N-2} & b_{N-2} & c_{N-2} \\ \alpha & & & \cdots & 0 & a_{N-1} & b_{N-1} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ \cdots \\ x_{N-2} \\ x_{N-1} \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \\ \cdots \\ r_{N-2} \\ r_{N-1} \end{bmatrix}$$

$$\mathbf{u} = \begin{bmatrix} \gamma \\ 0 \\ \vdots \\ 0 \\ \alpha \end{bmatrix} \qquad \mathbf{v} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ \beta/\gamma \end{bmatrix} \qquad b_0' = b_0 - \gamma, \qquad b_{N-1}' = b_{N-1} - \alpha\beta/\gamma$$

$$\begin{bmatrix} b_0 & c_0 & 0 & \cdots & & & \\ a_1 & b_1 & c_1 & \cdots & & & \\ & & & \cdots & & & \\ & & & \cdots & a_{N-2} & b_{N-2} & c_{N-2} \\ & & & \cdots & 0 & a_{N-1} & b_{N-1} \end{bmatrix} \cdot \begin{bmatrix} u_0 \\ u_1 \\ \cdots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \\ \cdots \\ r_{N-2} \\ r_{N-1} \end{bmatrix}$$

$\mathbf{A} \;\rightarrow\; (\mathbf{A} + \mathbf{u} \otimes \mathbf{v})$ where A is tridiagonal

Generalization: Woodbury formula

- Successive application of Sherman-Morrison to rank P, with $P << N$

$$(\mathbf{A} + \mathbf{U} \cdot \mathbf{V}^T)^{-1} = \mathbf{A}^{-1} - \left[\mathbf{A}^{-1} \cdot \mathbf{U} \cdot (\mathbf{1} + \mathbf{V}^T \cdot \mathbf{A}^{-1} \cdot \mathbf{U})^{-1} \cdot \mathbf{V}^T \cdot \mathbf{A}^{-1}\right]$$

- *U* and *V* are now *NxP* matrices
- Proof same as for Sherman-Morrison

Inversion by Partitioning

- Sometimes we can decompose the matrix into block sub-matrices P (dimension p x p), S (s x s) and Q and R (p x s and s x p)

$$\mathbf{A} = \begin{bmatrix} \mathbf{P} & \mathbf{Q} \\ \mathbf{R} & \mathbf{S} \end{bmatrix} \qquad \mathbf{A}^{-1} = \begin{bmatrix} \widetilde{\mathbf{P}} & \widetilde{\mathbf{Q}} \\ \widetilde{\mathbf{R}} & \widetilde{\mathbf{S}} \end{bmatrix}$$

$$\begin{aligned}
\widetilde{\mathbf{P}} &= (\mathbf{P} - \mathbf{Q} \cdot \mathbf{S}^{-1} \cdot \mathbf{R})^{-1} \\
\widetilde{\mathbf{Q}} &= -(\mathbf{P} - \mathbf{Q} \cdot \mathbf{S}^{-1} \cdot \mathbf{R})^{-1} \cdot (\mathbf{Q} \cdot \mathbf{S}^{-1}) \\
\widetilde{\mathbf{R}} &= -(\mathbf{S}^{-1} \cdot \mathbf{R}) \cdot (\mathbf{P} - \mathbf{Q} \cdot \mathbf{S}^{-1} \cdot \mathbf{R})^{-1} \\
\widetilde{\mathbf{S}} &= \mathbf{S}^{-1} + (\mathbf{S}^{-1} \cdot \mathbf{R}) \cdot (\mathbf{P} - \mathbf{Q} \cdot \mathbf{S}^{-1} \cdot \mathbf{R})^{-1} \cdot (\mathbf{Q} \cdot \mathbf{S}^{-1})
\end{aligned}$$

$$\det \mathbf{A} = \det \mathbf{P} \det(\mathbf{S} - \mathbf{R} \cdot \mathbf{P}^{-1} \cdot \mathbf{Q}) = \det \mathbf{S} \det(\mathbf{P} - \mathbf{Q} \cdot \mathbf{S}^{-1} \cdot \mathbf{R})$$

Vandermonde and Toeplitz Matrices

- Can be solved with N^2

- Vandermonde

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^{N-1} \\ 1 & x_1 & x_1^2 & \cdots & x_1^{N-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{N-1} & x_{N-1}^2 & \cdots & x_{N-1}^{N-1} \end{bmatrix} \cdot \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{N-1} \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{N-1} \end{bmatrix}$$

- Toeplitz

$$\begin{bmatrix} R_0 & R_{-1} & R_{-2} & \cdots & R_{-(N-2)} & R_{-(N-1)} \\ R_1 & R_0 & R_{-1} & \cdots & R_{-(N-3)} & R_{-(N-2)} \\ R_2 & R_1 & R_0 & \cdots & R_{-(N-4)} & R_{-(N-3)} \\ \cdots & & & \cdots & & \\ R_{N-2} & R_{N-3} & R_{N-4} & \cdots & R_0 & R_{-1} \\ R_{N-1} & R_{N-2} & R_{N-3} & \cdots & R_1 & R_0 \end{bmatrix}$$

Summary

- Linear algebra allows us to solve linear systems of equations, compute inverse and determinant of a matrix…
- N^3 scaling is very steep: we cannot do it above $N = 10^4$-10^5
- For larger dimensions iterative methods are needed: these will be discussed when we discuss optimization

Literature

- *Numerical Recipes*, Press et al., Chapter 2, 11
- *Computational Physics,* M. Newman, Chapter 6