Algorithms for Structured Matrices

Multiplying two arbitrary matrices

$$\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}\begin{bmatrix} d_1 & d_2 & d_3 \\ e_1 & e_2 & e_3 \\ f_1 & f_2 & f_3 \end{bmatrix} = \begin{bmatrix} a_1 d_1 + a_2 e_1 + a_3 f_1 & . & . \\ . & . & . \\ . & . & . \end{bmatrix}$$

This operation requires 27 multiplications and 18 adds andis $O(n^3)$.

Embarrassingly Simple Structured Matrix (ESSM)

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 5 & 10 & 15 \end{bmatrix}$$

Recall that the rank of a matrix is the number of linearly independent rows (or columns).

The rank of the ESSM is 1. It can be decomposed (actually compressed, as fewer terms are needed) as

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 5 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Decomposition takes linear time since assume we have foreknowledge that this is a rank 1 matrix.

Now suppose we want to multiply our ESSM by an arbitrary matrix M:

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 5 & 10 & 15 \end{bmatrix}$$

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \left(\begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 5 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)$$

$$\left(\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 2 & 0 & 0 \\ 5 & 0 & 0 \end{bmatrix} \right) \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The operation in the parentheses costs 9 multiplications and 6 adds and results in a matrix with the second and third columns zero:

$$\begin{bmatrix} m_{11}+m_{12}+m_{13} & 0 & 0 \\ 2m_{11}+2m_{12}+2m_{13} & 0 & 0 \\ 5m_{11}+5m_{12}+5m_{13} & 0 & 0 \end{bmatrix}$$

The final matrix multiplication (decompression) requires 9 multiplications and no adds for a total of 18 multiplications and 6 adds.

This algorithm is $O(n^2)$.

Compress

Operate

Decompress

To review,

A Toeplitz matrix: $\begin{bmatrix} a & b & c \\ e & a & b \\ d & e & a \end{bmatrix}$ (2*n* – 1 terms)

A circulant matrix: $\begin{bmatrix} a & b & c \\ c & a & b \\ b & c & a \end{bmatrix}$ (*n* terms)

Although these matrices are highly redundant we can't use the trick from the ESSM because all rows and columns are still linearly independent! What can we do?

Key observation:

With Toeplitz matrices, shifting the rows and columns down and to the right and superimposing on the original matrix does not change the entries. One would hope that this property could be exploited in some way to simplify calculations with these matrices. This is the concept of *displacement rank.*

We can investigate this with the *Sylvester operator*,

$$\nabla_{A,B} = AM - MB$$

Which is useful in compressing the various types of structured matrices *M* with appropriate choices of operator matrices *A* and *B*. For a circulant matrix we use $\nabla_{Z_1,Z_0} = Z_1M - MZ_0$

where $Z_0 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ $Z_1 = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

Z_0M shifts *M* down Z_1M rotates *M* down

MZ_0 shifts *M* left MZ_1 rotates *M* left

If we apply the transformation $Z_1M - MZ_0$ to the general

Toeplitz matrix $\begin{bmatrix} a & b & c \\ e & a & b \\ d & e & a \end{bmatrix}$ we get:

$$\begin{bmatrix} d & e & a \\ a & b & c \\ e & a & b \end{bmatrix} - \begin{bmatrix} b & c & 0 \\ a & b & 0 \\ e & a & 0 \end{bmatrix} = \begin{bmatrix} d-b & e-c & a \\ 0 & 0 & c \\ 0 & 0 & b \end{bmatrix}$$

This is a rank 2 matrix, meaning that the general Toeplitz matrix of displacement rank 2.

For a circulant matrix the same calculation gives us

$$\begin{bmatrix} b & c & a \\ a & b & c \\ c & a & b \end{bmatrix} - \begin{bmatrix} b & c & 0 \\ a & b & 0 \\ c & a & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & a \\ 0 & 0 & c \\ 0 & 0 & b \end{bmatrix}$$

for a displacement rank of 1.

Observe that the matrix on the right can be expressed as

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

In general, for matrices of displacement rank α, there exist matrices *G* and *B* such that

$$\nabla_{A,B} = AM - MB = GB$$

where *G* and *B* are $n \times \alpha$ and $\alpha \times n$ respectively.

Calculations on the generators are $O(n^2)$ as opposed to $O(n^3)$ for matrix-matrix calculations.

Circulant matrices have the additional property that they are diagonalized by the Fast Fourier Transform. That is, if C is a circulant matrix, then $F^{-1}CF$ is a diagonal matrix.

A simple example is

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}a & b\\ b & a\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \begin{bmatrix}a+b & 0\\ 0 & a-b\end{bmatrix}$$

This diagonal matrix can be calculated in linear time. The operation is invertible, so

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}a & b\\ b & a\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \begin{bmatrix}a & b\\ b & a\end{bmatrix}$$

Suppose we want to multiply an arbitrary vector by a circulant matrix, e.g. we want to calculate something like

$$\begin{bmatrix}a & b\\ b & a\end{bmatrix}\begin{bmatrix}2\\ 3\end{bmatrix} = \begin{bmatrix}2a+3b\\ 3b+2a\end{bmatrix}$$

which is O(n^2) done conventionally.

This product is equivalent to

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}a & b\\b & a\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\right)\left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}2\\3\end{bmatrix}\right)$$

The grouping $\left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}2\\3\end{bmatrix}\right)$ is a Fourier transform and can be computed in $n\log(n)$ time. The other grouping is a diagonal matrix that is calculated in linear time. So this equation reduces to

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\left(\begin{bmatrix}a+b & 0\\0 & a-b\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}5\\-1\end{bmatrix}\right)$$

The grouping $\left(\begin{bmatrix}a+b & 0\\0 & a-b\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}5\\-1\end{bmatrix}\right)$ is done in linear time, leaving $\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}5a+5b\\b-a\end{bmatrix}$ which is again a Fourier transform costing $n\log(n)$. Our final result is $\begin{bmatrix}2a+3b\\3a+2b\end{bmatrix}$ which happily is the same result we got doing the calculation directly.

A general n x n Toeplitz matrix can be converted into a $2n$ x $2n$ circulant matrix:

$$\begin{bmatrix} a & b & c \\ d & a & b \\ e & d & b \end{bmatrix} \Rightarrow \begin{bmatrix} a & b & c & 0 & d & e \\ e & a & b & c & 0 & d \\ d & e & a & b & c & 0 \\ 0 & d & e & a & b & c \\ c & 0 & d & e & a & b \\ b & c & 0 & d & e & a \end{bmatrix}$$

which means that it too can be diagonalized by the Fourier transform, allowing $O(n^3)$ calculations to be done in $O(n^2)$ time.