Структурированные матрицы, БПФ, свертки, матрицы Тёплица

Даня Меркулов

МФТИ. AI360

Структурированные матрицы

В предыдущих сериях

- До этого мы рассматривали один класс структурированных матриц: **разреженные матрицы**

В предыдущих сериях

- До этого мы рассматривали один класс структурированных матриц: **разреженные матрицы**
- Но итерационные методы работают хорошо для любой матрицы, которая имеет быстрый **black-box** матрично-векторный продукт

В предыдущих сериях

- До этого мы рассматривали один класс структурированных матриц: **разреженные матрицы**
- Но итерационные методы работают хорошо для любой матрицы, которая имеет быстрый **black-box** матрично-векторный продукт
- Важным классом таких матриц являются **матрицы Тёплица** (и **матрицы Ганкеля**) и их **многоуровневые** варианты

В предыдущих сериях

- До этого мы рассматривали один класс структурированных матриц: **разреженные матрицы**
- Но итерационные методы работают хорошо для любой матрицы, которая имеет быстрый **black-box** матрично-векторный продукт
- Важным классом таких матриц являются **матрицы Тёплица** (и **матрицы Ганкеля**) и их **многоуровневые** варианты
- Они непосредственно связаны с **операцией свертки** и **БПФ**.

Свертка

- Одна из ключевых операций в обработке сигналов/машинном обучении --- это **свертка двух функций**.

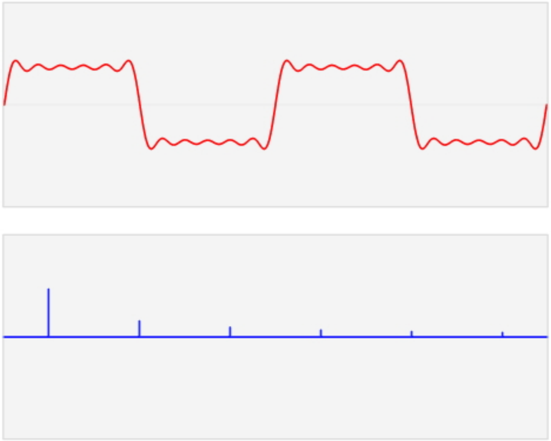
Свертка

- Одна из ключевых операций в обработке сигналов/машинном обучении --- это **свертка двух функций**.
- Пусть $x(t)$ и $y(t)$ --- две заданные функции. Их свертка определяется как

$$(x * y)(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau.$$

Теорема свертки и преобразование Фурье

- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.



Теорема свертки и преобразование Фурье

- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$
\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt} x(t) dt.
$$

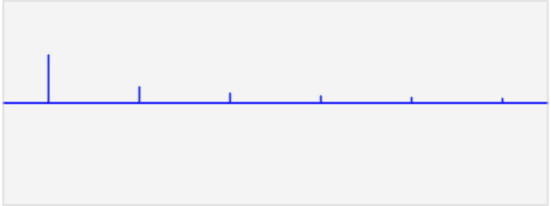
Теорема свертки и преобразование Фурье

- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt} x(t) dt.$$

- Тогда,

$$\mathcal{F}(x * y) = \mathcal{F}(x)\mathcal{F}(y).$$



Теорема свертки и преобразование Фурье

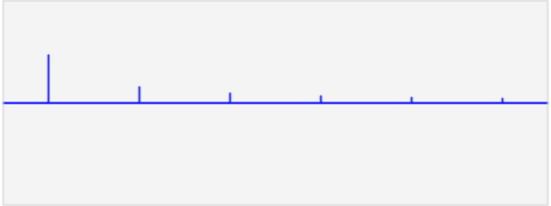
- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt} x(t) dt.$$

- Тогда,

$$\mathcal{F}(x * y) = \mathcal{F}(x)\mathcal{F}(y).$$

- Таким образом, алгоритм для вычисления свертки может быть следующим:



Теорема свертки и преобразование Фурье

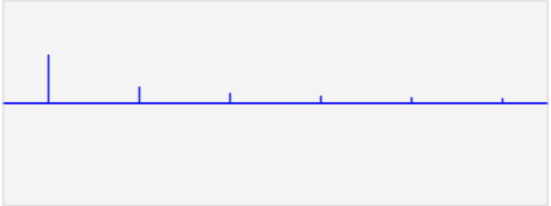
- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt} x(t) dt.$$

- Тогда,

$$\mathcal{F}(x * y) = \mathcal{F}(x)\mathcal{F}(y).$$

- Таким образом, алгоритм для вычисления свертки может быть следующим:
    1. Вычислить преобразование Фурье от $x(t)$ и $y(t)$.



Теорема свертки и преобразование Фурье

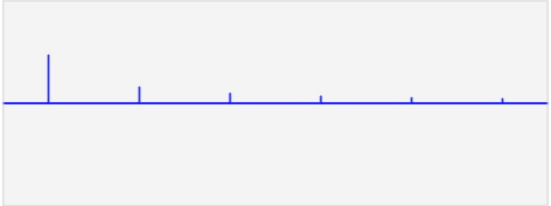
- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt} x(t) dt.$$

- Тогда,

$$\mathcal{F}(x * y) = \mathcal{F}(x)\mathcal{F}(y).$$

- Таким образом, алгоритм для вычисления свертки может быть следующим:
    1. Вычислить преобразование Фурье от $x(t)$ и $y(t)$.
    2. Вычислить их произведение



Теорема свертки и преобразование Фурье

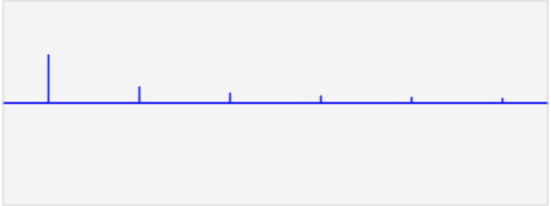
- Известный факт: **свертка** в **временной области** эквивалентна **произведению** в **частотной области**.
- Преобразование Фурье дает соответствие между временной и частотной областями:

$$\hat{x}(w) = (\mathcal{F}(x))(w) = \int_{-\infty}^{\infty} e^{iwt}x(t)dt.$$

- Тогда,

$$\mathcal{F}(x * y) = \mathcal{F}(x)\mathcal{F}(y).$$

- Таким образом, алгоритм для вычисления свертки может быть следующим:
    1. Вычислить преобразование Фурье от $x(t)$ и $y(t)$.
    2. Вычислить их произведение
    3. Вычислить обратное преобразование Фурье



Дискретная свертка

$$(x * y)(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau.$$

Приблизим интеграл по квадратурной сумме на равномерной сетке и сохраним сигнал в равномерных точках.

Тогда останется суммирование

$$z_i = \sum_{j=0}^{n-1} x_j y_{i-j},$$

которое называется **дискретной сверткой**. Это можно рассматривать как применение фильтра с коэффициентами x к сигналу y.

Существуют разные фильтры для разных целей, но все они используют **сдвиговую инвариантность**.

Дискретная свертка и матрицы Тёплица

Дискретная свертка может быть представлена как матрично-векторное произведение:

$$z_i = \sum_{j=0}^{n-1} x_j y_{i-j}, \Leftrightarrow z = Ax$$

где элементы матрицы A задаются как $a_{ij} = y_{i-j}$, т.е. они зависят только от разности между индексом строки и столбца.

Такие матрицы называются **матрицами Тёплица**.

Определение матриц Тёплица

Матрица называется **матрицей Тёплица**, если ее элементы определяются как

$$a_{ij} = t_{i-j}.$$

- Матрица Тёплица полностью определяется своим первым столбцом и первой строкой (т.е. $2n-1$ параметров).

Определение матриц Тёплица

Матрица называется **матрицей Тёплица**, если ее элементы определяются как

$$a_{ij} = t_{i-j}.$$

- Матрица Тёплица полностью определяется своим первым столбцом и первой строкой (т.е. $2n-1$ параметров).
- Она является **плотной матрицей**, однако является **структурированной матрицей** (т.е. определяется $\mathcal{O}(n)$ параметрами).

Определение матриц Тёплица

Матрица называется **матрицей Тёплица**, если ее элементы определяются как

$$a_{ij} = t_{i-j}.$$

- Матрица Тёплица полностью определяется своим первым столбцом и первой строкой (т.е. $2n-1$ параметров).
- Она является **плотной матрицей**, однако является **структурированной матрицей** (т.е. определяется $\mathcal{O}(n)$ параметрами).
- Основная операция в дискретной свертке --- это произведение матрицы Тёплица на вектор.

Определение матриц Тёплица

Матрица называется **матрицей Тёплица**, если ее элементы определяются как

$$a_{ij} = t_{i-j}.$$

- Матрица Тёплица полностью определяется своим первым столбцом и первой строкой (т.е. $2n - 1$ параметров).
- Она является **плотной матрицей**, однако является **структурированной матрицей** (т.е. определяется $\mathcal{O}(n)$ параметрами).
- Основная операция в дискретной свертке --- это произведение матрицы Тёплица на вектор.
- Можно ли вычислить его быстрее, чем за $\mathcal{O}(n^2)$?

Матрицы Тёплица и циркулянты

- Для специального класса матриц Тёплица, называемых **циркулянтами**, можно вычислить произведение матрицы на вектор быстрее, чем за $\mathcal{O}(n^2)$.

Матрицы Тёплица и циркулянты

- Для специального класса матриц Тёплица, называемых **циркулянтами**, можно вычислить произведение матрицы на вектор быстрее, чем за $\mathcal{O}(n^2)$.
- Матрица C называется **циркулянтом**, если

$$C_{ij} = c_{i-j \mod n},$$

т.е. она периодически заворачивается.

$$C = \begin{bmatrix} c_0 & c_3 & c_2 & c_1 \\ c_1 & c_0 & c_3 & c_2 \\ c_2 & c_1 & c_0 & c_3 \\ c_3 & c_2 & c_1 & c_0 \end{bmatrix}.$$

Матрицы Тёплица и циркулянты

- Для специального класса матриц Тёплица, называемых **циркулянтами**, можно вычислить произведение матрицы на вектор быстрее, чем за $\mathcal{O}(n^2)$.
- Матрица C называется **циркулянтом**, если

$$C_{ij} = c_{i-j \mod n},$$

т.е. она периодически заворачивается.

$$C = \begin{bmatrix} c_0 & c_3 & c_2 & c_1 \\ c_1 & c_0 & c_3 & c_2 \\ c_2 & c_1 & c_0 & c_3 \\ c_3 & c_2 & c_1 & c_0 \end{bmatrix}.$$

- Циркулянты имеют одинаковые **собственные векторы**, задаваемые дискретным преобразованием Фурье (DFT).

Теорема о спектральном разложении для циркулянтов

Теорема:

Любая циркулянтная матрица может быть представлена в виде

$$C = \frac{1}{n} F^* \Lambda F,$$

где F --- это **матрица Фурье** с элементами

$$F_{kl} = w_n^{kl}, \quad k, l = 0, \dots, n-1, \quad w_n = e^{-\frac{2\pi i}{n}},$$

и матрица $\Lambda = \text{diag}(\lambda)$ --- это диагональная матрица и

$$\lambda = Fc,$$

где c --- это первый столбец циркулянтной матрицы C.

Матрица Фурье

Матрица Фурье определяется как:

$$F_n = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w_n^{1\cdot 1} & w_n^{1\cdot 2} & \dots & w_n^{1\cdot (n-1)} \\ 1 & w_n^{2\cdot 1} & w_n^{2\cdot 2} & \dots & w_n^{2\cdot (n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & w_n^{(n-1)\cdot 1} & w_n^{(n-1)\cdot 2} & \dots & w_n^{(n-1)\cdot (n-1)} \end{pmatrix},$$

или эквивалентно

$$F_n = \{w_n^{kl}\}_{k,l=0}^{n-1},$$

где

$$w_n = e^{-\frac{2\pi i}{n}}.$$

Свойства:

- Симметричная (не эрмитова!)

Матрица Фурье

Матрица Фурье определяется как:

$$F_n = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w_n^{1\cdot 1} & w_n^{1\cdot 2} & \dots & w_n^{1\cdot (n-1)} \\ 1 & w_n^{2\cdot 1} & w_n^{2\cdot 2} & \dots & w_n^{2\cdot (n-1)} \\ \dots & \dots & \dots & \dots & \dots \\ 1 & w_n^{(n-1)\cdot 1} & w_n^{(n-1)\cdot 2} & \dots & w_n^{(n-1)\cdot (n-1)} \end{pmatrix},$$

или эквивалентно

$$F_n = \{w_n^{kl}\}_{k,l=0}^{n-1},$$

где

$$w_n = e^{-\frac{2\pi i}{n}}.$$

Свойства:

- Симметричная (не эрмитова!)
- Унитарная с точностью до масштабирования: $F_n^* F_n = F_n F_n^* = nI$ (проверьте этот факт). Следовательно, $F_n^{-1} = \frac{1}{n} F_n^*$

Матрица Фурье

Матрица Фурье определяется как:

$$
F_n = \begin{pmatrix}
1 & 1 & 1 & \dots & 1 \\
1 & w_n^{1\cdot 1} & w_n^{1\cdot 2} & \dots & w_n^{1\cdot (n-1)} \\
1 & w_n^{2\cdot 1} & w_n^{2\cdot 2} & \dots & w_n^{2\cdot (n-1)} \\
\dots & \dots & \dots & \dots & \dots \\
1 & w_n^{(n-1)\cdot 1} & w_n^{(n-1)\cdot 2} & \dots & w_n^{(n-1)\cdot (n-1)}
\end{pmatrix},
$$

или эквивалентно

$$
F_n = \{w_n^{kl}\}_{k,l=0}^{n-1},
$$

где

$$
w_n = e^{-\frac{2\pi i}{n}}.
$$

Свойства:

- Симметричная (не эрмитова!)
- Унитарная с точностью до масштабирования: $F_n^* F_n = F_n F_n^* = nI$ (проверьте этот факт). Следовательно, $F_n^{-1} = \frac{1}{n} F_n^*$
- Можно умножать на вектор (называемый дискретным преобразованием Фурье или DFT) с $\mathcal{O}(n \log n)$ сложностью (называемый быстрым преобразованием Фурье или FFT)!

Матрица Фурье

Матрица Фурье определяется как:

$$
F_n = \begin{pmatrix}
1 & 1 & 1 & \dots & 1 \\
1 & w_n^{1\cdot 1} & w_n^{1\cdot 2} & \dots & w_n^{1\cdot (n-1)} \\
1 & w_n^{2\cdot 1} & w_n^{2\cdot 2} & \dots & w_n^{2\cdot (n-1)} \\
\dots & \dots & \dots & \dots & \dots \\
1 & w_n^{(n-1)\cdot 1} & w_n^{(n-1)\cdot 2} & \dots & w_n^{(n-1)\cdot (n-1)}
\end{pmatrix},
$$

или эквивалентно

$$
F_n = \{w_n^{kl}\}_{k,l=0}^{n-1},
$$

где

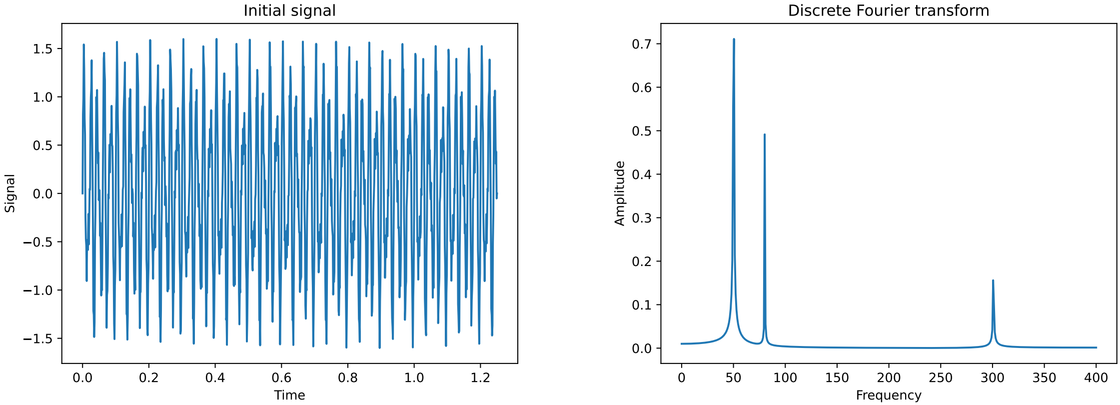
$$
w_n = e^{-\frac{2\pi i}{n}}.
$$

Свойства:

- Симметричная (не эрмитова!)
- Унитарная с точностью до масштабирования: $F_n^* F_n = F_n F_n^* = nI$ (проверьте этот факт). Следовательно, $F_n^{-1} = \frac{1}{n} F_n^*$
- Можно умножать на вектор (называемый дискретным преобразованием Фурье или DFT) с $\mathcal{O}(n \log n)$ сложностью (называемый быстрым преобразованием Фурье или FFT)!
- FFT помогает анализировать спектр сигнала и, как мы увидим позже, помогает выполнять быстрые умножения с определенными типами матриц.

Дискретное преобразование Фурье

$$f(x) = \sin(50.0 \cdot 2\pi x) + 0.5 \cdot \sin(80.0 \cdot 2\pi x) + 0.2 \cdot \sin(300.0 \cdot 2\pi x)$$



Быстрое преобразование Фурье (FFT/БПФ)

Рассмотрим матричное представление стандартного алгоритма Cooley-Tukey (1965), основная идея которого --- **разделяй и властвуй**.

Обратите внимание, что в пакетах используются более сложные версии.

- Пусть n --- степень двойки.

Быстрое преобразование Фурье (FFT/БПФ)

Рассмотрим матричное представление стандартного алгоритма Cooley-Tukey (1965), основная идея которого --- **разделяй и властвуй**.

Обратите внимание, что в пакетах используются более сложные версии.

- Пусть n --- степень двойки.
- Сначала мы **переставляем строки** матрицы Фурье так, что первые $n/2$ строк новой матрицы имеют номера $1, 3, 5, \dots, n-1$ и последние $n/2$ строки имеют номера $2, 4, 6 \dots, n$.

Быстрое преобразование Фурье (FFT/БПФ)

Рассмотрим матричное представление стандартного алгоритма Cooley-Tukey (1965), основная идея которого --- **разделяй и властвуй**.

Обратите внимание, что в пакетах используются более сложные версии.

- Пусть n --- степень двойки.
- Сначала мы **переставляем строки** матрицы Фурье так, что первые $n/2$ строк новой матрицы имеют номера $1, 3, 5, \dots, n-1$ и последние $n/2$ строки имеют номера $2, 4, 6 \dots, n$.
- Это можно выразить как умножение на перестановочную матрицу P_n:

$$
P_n = \left(\begin{array}{ccccccc}
1 & 0 & 0 & 0 & \dots & 0 & 0 \\
0 & 0 & 1 & 0 & \dots & 0 & 0 \\
\vdots & & & & & & \vdots \\
0 & 0 & 0 & 0 & \dots & 1 & 0 \\
\hline
0 & 1 & 0 & 0 & \dots & 0 & 0 \\
0 & 0 & 0 & 1 & \dots & 0 & 0 \\
\vdots & & & & & & \vdots \\
0 & 0 & 0 & 0 & \dots & 0 & 1
\end{array}\right),
$$

Быстрое преобразование Фурье (FFT/БПФ)

Таким образом,

$$
P_nF_n = \left(\begin{array}{ccccc}
1 & 1 & 1 & \dots & 1 \\
1 & w_n^{2\cdot 1} & w_n^{2\cdot 2} & \dots & w_n^{2\cdot (n-1)} \\
1 & w_n^{4\cdot 1} & w_n^{4\cdot 2} & \dots & w_n^{4\cdot (n-1)} \\
\vdots & & & & \vdots \\
1 & w_n^{(n-2)\cdot 1} & w_n^{(n-2)\cdot 2} & \dots & w_n^{(n-2)\cdot (n-1)} \\
\hline
1 & w_n^{1\cdot 1} & w_n^{1\cdot 2} & \dots & w_n^{1\cdot (n-1)} \\
1 & w_n^{3\cdot 1} & w_n^{3\cdot 2} & \dots & w_n^{3\cdot (n-1)} \\
\vdots & & & & \vdots \\
1 & w_n^{(n-1)\cdot 1} & w_n^{(n-1)\cdot 2} & \dots & w_n^{(n-1)\cdot (n-1)}
\end{array}\right),
$$

Теперь представим, что мы разделили её столбцы и строки на две части размера $n/2$.

Быстрое преобразование Фурье (FFT/БПФ)

В результате мы получаем 2×2 **блочную матрицу**, которая имеет следующий вид

$$P_n F_n = \begin{pmatrix} \left\{w_n^{2kl}\right\} & \left\{w_n^{2k(\frac{n}{2}+l)}\right\} \\ \left\{w_n^{(2k+1)l}\right\} & \left\{w_n^{(2k+1)(\frac{n}{2}+l)}\right\} \end{pmatrix}, \quad k, l = 0, \dots, \frac{n}{2} - 1.$$

Пока что это не выглядит как что-то, что работает быстрее :) Но мы увидим, что это так. Давайте рассмотрим первый блок $\{w_n^{2kl}\}$:

$$w_n^{2kl} = e^{-2kl\frac{2\pi i}{n}} = e^{-kl\frac{2\pi i}{n/2}} = w_{n/2}^{kl}.$$

Таким образом, этот блок является в точности матрицей Фурье в два раза меньшего размера $F_{n/2}$!

Блок $\left\{w_n^{(2k+1)l}\right\}$ можно записать как

$$w_n^{(2k+1)l} = w_n^{2kl+l} = w_n^l w_n^{2kl} = w_n^l w_{n/2}^{kl},$$

который можно записать как $W_{n/2} F_{n/2}$, где

$$W_{n/2} = \mathrm{diag}(1, w_n, w_n^2, \dots, w_n^{n/2-1}).$$

Быстрое преобразование Фурье (FFT/БПФ)

Проделав те же самые трюки для других блоков, мы, наконец, получим

$$P_n F_n = \begin{pmatrix} F_{n/2} & F_{n/2} \\ F_{n/2} W_{n/2} & -F_{n/2} W_{n/2} \end{pmatrix} = \begin{pmatrix} F_{n/2} & 0 \\ 0 & F_{n/2} \end{pmatrix} \begin{pmatrix} I_{n/2} & I_{n/2} \\ W_{n/2} & -W_{n/2} \end{pmatrix}.$$

- Таким образом, мы **уменьшили умножение на F_n до 2 умножений на $F_{n/2}$** и дешевых умножений на диагональные матрицы.

Быстрое преобразование Фурье (FFT/БПФ)

Проделав те же самые трюки для других блоков, мы, наконец, получим

$$P_n F_n = \begin{pmatrix} F_{n/2} & F_{n/2} \\ F_{n/2} W_{n/2} & -F_{n/2} W_{n/2} \end{pmatrix} = \begin{pmatrix} F_{n/2} & 0 \\ 0 & F_{n/2} \end{pmatrix} \begin{pmatrix} I_{n/2} & I_{n/2} \\ W_{n/2} & -W_{n/2} \end{pmatrix}.$$

- Таким образом, мы **уменьшили умножение на F_n до 2 умножений на $F_{n/2}$** и дешевых умножений на диагональные матрицы.
- Если мы применим полученные выражения рекурсивно к $F_{n/2}$, мы получим $\mathcal{O}(n \log n)$ сложность.

Циркулянтные матрицы

БПФ помогает быстро умножать на определенные типы матриц. Начнем с циркулянтной матрицы:

$$
C = \begin{pmatrix} c_0 & c_{n-1} & c_{n-2} & \dots & c_1 \\ c_1 & c_0 & c_{n-1} & \dots & c_2 \\ c_2 & c_1 & c_0 & \dots & c_3 \\ \dots & \dots & \dots & \dots & \dots \\ c_{n-1} & c_{n-2} & c_{n-3} & \dots & c_0 \end{pmatrix}
$$

Теорема. Пусть C --- циркулянтная матрица размера $n \times n$ и пусть c --- ее первый столбец, тогда

$$
C = \frac{1}{n} F_n^* \text{diag}(F_n c) F_n
$$

Быстрый матвек с циркулянтной матрицей

- Представление $C = \frac{1}{n} F^* \text{diag}(F_n c) F_n$ дает нам явный способ умножать вектор x на C за $\mathcal{O}(n \log n)$ операций.

Быстрый матвек с циркулянтной матрицей

- Представление $C = \frac{1}{n}F^* \text{diag}(F_n c)F_n$ дает нам явный способ умножать вектор x на C за $\mathcal{O}(n \log n)$ операций.
- Действительно,

$$Cx = \frac{1}{n}F_n^* \text{diag}(F_n c)F_n x = \text{ifft}\left(\text{fft}(c) \circ \text{fft}(x)\right)$$

где $\circ$ обозначает поэлементное произведение (Hadamard product) двух векторов (так как $\text{diag}(a)b = a \circ b$) и `ifft` обозначает обратное дискретное преобразование Фурье F_n^{-1}.

Матрицы Теплица

Теперь вернемся к матрицам Тёплица!

$$
T = \begin{pmatrix}
t_0 & t_{-1} & t_{-2} & t_{-3} & \dots & t_{1-n} \\
t_1 & t_0 & t_{-1} & t_{-2} & \dots & t_{2-n} \\
t_2 & t_1 & t_0 & t_{-1} & \dots & t_{3-n} \\
t_3 & t_2 & t_1 & t_0 & \dots & t_{4-n} \\
\dots & \dots & \dots & \dots & \dots & \dots \\
t_{n-1} & t_{n-2} & t_{n-3} & t_{n-4} & \dots & t_0
\end{pmatrix},
$$

или эквивалентно $T_{ij} = t_{i-j}$.

Операция матрично-векторного произведения может быть записана как

$$
y_i = \sum_{j=1}^{n} t_{i-j} x_j,
$$

которая может быть интерпретирована как дискретная **свертка** фильтра t_i и сигнала x_i.

Для простоты размер фильтра t таков, что размер входного и выходного сигналов одинаковый.

Обычно размер фильтра может быть произвольным.

Быстрая свертка имеет множество приложений, например, в обработке сигналов или в дифференциальных и интегральных уравнениях.

Быстрый матвек с матрицами Тёплица

Key point: the multiplication by a Toeplitz matrix can be reduced to the multiplication by a circulant.

- Indeed, every Toeplitz matrix of size $n \times n$ can be embedded into a Circulant matrix C of size $(2n-1) \times (2n-1)$:

$$C = \begin{pmatrix} T & \dots \\ \dots & \dots \end{pmatrix}.$$

Быстрый матвек с матрицами Тёплица

Key point: the multiplication by a Toeplitz matrix can be reduced to the multiplication by a circulant.

- Indeed, every Toeplitz matrix of size $n \times n$ can be embedded into a Circulant matrix C of size $(2n-1) \times (2n-1)$:

$$C = \begin{pmatrix} T & \dots \\ \dots & \dots \end{pmatrix}.$$

- The 3×3 matrix $T = \begin{pmatrix} t_0 & t_{-1} & t_{-2} \\ t_1 & t_0 & t_{-1} \\ t_2 & t_1 & t_0 \end{pmatrix}$ can be embedded as follows

$$C = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & t_2 & t_1 \\ t_1 & t_0 & t_{-1} & t_{-2} & t_2 \\ t_2 & t_1 & t_0 & t_{-1} & t_{-2} \\ t_{-2} & t_2 & t_1 & t_0 & t_{-1} \\ t_{-1} & t_{-2} & t_2 & t_1 & t_0 \end{pmatrix}.$$

Быстрый матвек с матрицами Тёплица

- For matvec $\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} \\ t_1 & t_0 & t_{-1} \\ t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$ we pad vector x with zeros:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \star \\ \star \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & t_2 & t_1 \\ t_1 & t_0 & t_{-1} & t_{-2} & t_2 \\ t_2 & t_1 & t_0 & t_{-1} & t_{-2} \\ t_{-2} & t_2 & t_1 & t_0 & t_{-1} \\ t_{-1} & t_{-2} & t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix} = \text{ifft}(\text{fft}(\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_{-2} \\ t_{-1} \end{pmatrix}) \circ \text{fft}(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix})).$$

Быстрый матвек с матрицами Тёплица

- For matvec $\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} \\ t_1 & t_0 & t_{-1} \\ t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$ we pad vector x with zeros:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \star \\ \star \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & t_2 & t_1 \\ t_1 & t_0 & t_{-1} & t_{-2} & t_2 \\ t_2 & t_1 & t_0 & t_{-1} & t_{-2} \\ t_{-2} & t_2 & t_1 & t_0 & t_{-1} \\ t_{-1} & t_{-2} & t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix} = \mathrm{ifft}(\mathrm{fft}(\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_{-2} \\ t_{-1} \end{pmatrix}) \circ \mathrm{fft}(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix})).$$

- Note that you **do not need to form and store** the whole matrix T

Быстрый матвек с матрицами Тёплица

- For matvec $\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} \\ t_1 & t_0 & t_{-1} \\ t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$ we pad vector x with zeros:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \star \\ \star \end{pmatrix} = \begin{pmatrix} t_0 & t_{-1} & t_{-2} & t_2 & t_1 \\ t_1 & t_0 & t_{-1} & t_{-2} & t_2 \\ t_2 & t_1 & t_0 & t_{-1} & t_{-2} \\ t_{-2} & t_2 & t_1 & t_0 & t_{-1} \\ t_{-1} & t_{-2} & t_2 & t_1 & t_0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix} = \mathrm{ifft}(\mathrm{fft}(\begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_{-2} \\ t_{-1} \end{pmatrix}) \circ \mathrm{fft}(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 0 \\ 0 \end{pmatrix})).$$

- Note that you **do not need to form and store** the whole matrix T
- From the Cooley-Tukey algorithm follows that the preferable size of circulant matrix is 2^k for some k. You can do it with zero padding of the appropriate size.