Prop. If $f$ is $C^k$, then $\hat{f}(\omega) = o\left(\frac{1}{\ln(\omega)}\right)$ as $\ln(\omega) \to \infty$.

If $\hat{f}(\omega) = O\left(\frac{1}{\ln(\omega + \epsilon)}\right)$ then $f$ is $C^k$.

Proof.

$$\sum_{n \in \mathbb{Z}} f(n) e^{-2\pi i n x} \text{ abs. conv.}$$

Differentiate by term by term:

$$\frac{d}{dx} \sum_{n \in \mathbb{Z}} f(n) e^{-2\pi i n x} \to \text{ sum is } C^k.$$

Can't just differentiate term by term:

$$\hat{f}'(\omega) = \int_0^1 f'(x) e^{-2\pi i \omega x} \, dx$$

$$= \int_0^1 \frac{f'(x)}{2\pi i \omega} e^{-2\pi i \omega x} \, dx$$

$$= \frac{\hat{f}'(\omega)}{2\pi i \omega}$$

$$\hat{f}(\omega) = O(2\pi i \omega^n \hat{f}(\omega))$$

$\to 0$ since $\hat{f}(\omega)$ cont.

QED.
Suppose \( f \) is \( C^\infty \) on \( \mathbb{R}(x) \).

\[
\frac{1}{N} \sum_{j=0}^{N-1} f\left( \frac{x}{N} \right) - \int f(x) \, dx = ?
\]

\[
\sum_{n \in \mathbb{Z}} \frac{1}{N} \sum_{j=0}^{N-1} e^{2\pi i n j/N} = \sum_{n \in \mathbb{Z}} \frac{f(n)}{N} \quad \text{for all } n \equiv 0 \pmod{N}
\]

\[
f(x) = \int f(x) \, dx
\]

\[
|\hat{f}(n)| = o\left( \frac{1}{\log n} \right) \quad \text{as } n \to \infty
\]

So get excellent conv. of Riemann sums

error term = \[
\sum_{n \in \mathbb{Z}} \frac{f(n)}{N}. \quad \text{wheels going backwards}
\]

**Aliasing:** if \( n \equiv m \pmod{N} \)

\[
e^{2\pi i n x} = e^{2\pi i m x} \quad \text{for all } x \in \mathbb{Z}
\]

\[
e^{2\pi i n x}, e^{2\pi i m x} \text{ cannot be dist. by } N \text{ equispaced samples}
\]

Riemann sums are best possible subject to aliasing
F.F.T.

Cordy-Tuckey Iberi Gauss story (Cont.)

want to compute
\[ \frac{1}{N} \sum_{k=0}^{N-1} f(\frac{k}{N}) e^{-2\pi i km/N} \]

to get \( \hat{f}(m) \) + aliased freq's

For \( g: \mathbb{Z}/N\mathbb{Z} \to \mathbb{C} \), define P.F.T.
\[ \hat{g}(m) = \frac{1}{N} \sum_{k=0}^{N-1} g(k) e^{-2\pi i \frac{mk}{N}} \]

\( \xi = e^{2\pi i/N} \)

\( (g,f) \) notation conflict.

\( \) but that's OK.

i.e., we are computing D.F.T. for \( g(x) = f(xN) \)

Fourier inversion:
\[ g(m) = \sum_{n=0}^{N-1} \hat{g}(n) \xi^{mn} \]

Proof:
\[ \sum_{n=0}^{N-1} \hat{g}(n) \xi^{mn} = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{k=0}^{N-1} g(k) \xi^{-kn} \xi^{mn} \]
\[ = \frac{1}{N} \sum_{k=0}^{N-1} g(k) \sum_{n=0}^{N-1} \xi^{mk-n} \]
\[ = g(m) \]

Q.E.D.
representation theory of \( \mathbb{Z}/N\mathbb{Z} \)

\[ x_n(n) = \sum_{m=0}^{\infty} \hat{g}(m) x_n \]

\[ g = \sum_{n=0}^{N-1} \hat{g}(n) x_n \]

so \( \hat{g} \) gives coefficients in terms of \( x_n \)

---

How hard is it to compute the D.F.T.? Count arithmetic operations:

- \( N^2 \) obvious (\( N \) pts., \( N \) ops. each)
- linear time? can't even read data faster

Fast Fourier Transform:

\( N \log N \) (exp. improvement per data pt.)

Open Q: can it be improved?

Trivial case: \( ab + ac \) vs. \( a(b+c) \)

reformulate:

\[ p(x) = \sum_{n} g(n) x^n \]

\[ p(\zeta^n) = \hat{g}(n) \]

coeffs of poly of deg < \( N \) \( \xrightarrow{\text{DFT}} \) values of poly at \( N \)th roots of 1

40kHz sampling so large #5

CD: 44.1kHz Hz
Cooky-Tukey: assume \( N = 2^L \)

\[
p(x) = g(x^2) + x^r(x^2)
\]

\[
\deg(p) < N \implies \deg(g), \deg(r) < \frac{N}{2}
\]

\[
S \text{ runs over } \frac{N}{2} \text{ roots of 1}
\]

Do FFTs on \( g, r \), then combine

running time:

\[
T_N \leq 2T_{N/2} + CN
\]

\[
\leq 4T_N/4 + 2C N/2 + CN
\]

\[
\leq 8T_N/8 + 4C N/4 + 2CN/2 + CN
\]

\[
\vdots
\]

\[
\leq 2^L T_1 + 2CN
\]

\[
= CN \log_2 N + T_1 N
\]

\[
= O(N \log^2 N)
\]

Fast poly (or int.) mult.

Kolmogorov: conv. \( N^2 \) opt.

Karatsuba: \( O(N \log_3 2) \)

Toom: \( O(N^{1 + o(1)}) \)
FFTr-based fast mult:

\[ \text{poly } \xrightarrow{\text{FFT}} \text{values at roots of } 1 \]

\[ \xrightarrow{\text{FFT}^{-1}} \]

just need \( N > \deg(\text{product}) \)

can choose \( N = \text{power of } 2 \)

get \( O(N \log N) \) complex ops to multiply \( \deg N \) polys / time.

More generally:

\[ (f \ast g)(m) = \frac{1}{N} \sum_{k=0}^{N-1} f(k) g(m-k) \]

\[ \hat{f} \ast \hat{g}(m) = \frac{1}{N^2} \sum_{k,k'} f(k) g(k-m) \]

\[ = \frac{1}{N^2} \sum_{k} f(k) \sum_{k'} g(k-m) \]

\[ = \frac{1}{N} \{ |f| \ast |g| \}(m) \]

So can do fast convolution (poly. mult is special case)

smoothly = Gaussian blur
filtering: remove freqs

cross corr: compute \( \sum_{k=0}^{N-1} f(k) g((m+k) \mod N) \) \( \forall m \)

\[ f \ast g \]
What if $N$ is not a power of 2?

Yes (Bluestein)

$$g(m) = \frac{1}{N} \sum_{n} g(n) \ z^{-mn}$$

separate $mn$ as much as possible

polarization:

$$mn = \frac{m^2 + n^2 - (m-n)^2}{2}$$

$$g(m) = \frac{z^{-m^{2}/2}}{N} \sum_{n} g(n) \ z^{-n^{2}/2} z^{-(m-n)^2/2}$$

Convolution of

coeffs of $x^m$ in

$$\left( \sum_{n=0}^{N-1} g(n) \ z^{-n^{2}/2} x^n \right) \left( \sum_{n=-(N-1)}^{N-1} z^{n^{2}/2} x^n \right)$$

Laurent poly product

can compute in $O(M\log N)$ time!

Why can this work?

Changing modulus for DFT.