A Tale of Two Norms
The Weyl norm

\[ f \in \mathcal{H}_d^F[1] \quad f = \sum_{|\alpha| = d} f_\alpha X^\alpha \]

where \( \alpha = (\alpha_0, \ldots, \alpha_n) \in \mathbb{N}^{n+1} \) and \(|\alpha| = \alpha_0 + \cdots + \alpha_n\).

\[ \|f\|_W := \sqrt{\sum_{|\alpha| = d} \binom{d}{\alpha}^{-1} |f_\alpha|^2} \]

where \( \binom{d}{\alpha} \) is the multinomial coefficient \( \frac{d!}{\alpha_0! \cdots \alpha_n!} \).

For \( f = (f_1, \ldots, f_q) \in \mathcal{H}_d[q] \) the Weyl norm extends as

\[ \|f\|_W := \sqrt{\|f_1\|_W^2 + \cdots + \|f_q\|_W^2} \]
The $\infty$ norm

$$\| f \|_\infty^F := \begin{cases} \max_{x \in S^n} \| f(x) \|_\infty = \max_{x \in S^n} \max_i |f_i(x)| & \text{if } F = \mathbb{R} \\ \max_{z \in \mathbb{P}^n} \| f(z) \|_\infty = \max_{z \in \mathbb{P}^n} \max_i |f_i(z)| & \text{if } F = \mathbb{C} \end{cases}$$

Why bother to choose $\| f \|_\infty^F$ over $\| f \|_W$?
Why bother?

Reason 1: There is a huge gain for random data!

In the worst-case,

\[ \|f\|_\infty^F \leq \|f\|_W \]

In the random case,

**Theorem**

*For random \( f \in \mathcal{H}_d^F[q] \),*

\[ \mathbb{E}_{f} \frac{\|f\|_\infty^F}{\|f\|_W} \leq \mathcal{O}\left(\sqrt{\frac{n \ln(eD)}{N}}\right) \sim \mathcal{O}\left(\sqrt{\frac{\ln(eD)}{D^n}}\right) \text{ (for large } D) \]

Huge gain for ‘typical’ input
Why bother?

Reason 2:
The $\infty$-norm can still control the derivatives!

<table>
<thead>
<tr>
<th>Theorem</th>
</tr>
</thead>
<tbody>
<tr>
<td>Let $F \in { \mathbb{R}, \mathbb{C} }$, $f \in \mathcal{H}_d^F[1]$, $x \in \mathbb{F}^{n+1}$ and $\nu \in \mathbb{F}^{n+1}$, then</td>
</tr>
<tr>
<td>$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Theorem (Kellogg’s Inequality)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Let $F \in { \mathbb{R}, \mathbb{C} }$, $f \in \mathcal{H}_d^F[1]$, $x \in \mathbb{F}^{n+1}$ and $\nu \in \mathbb{F}^{n+1}$, then</td>
</tr>
<tr>
<td>$</td>
</tr>
</tbody>
</table>
Why bother?

Reason 2:
The $\infty$-norm can still control the derivatives!

**Theorem**

Let $F \in \{\mathbb{R}, \mathbb{C}\}$, $f \in \mathcal{H}^F_d[1]$, $x \in \mathbb{F}^{n+1}$ and $v \in \mathbb{F}^{n+1}$, then

$$|\overrightarrow{D}_xf \cdot v| \leq d^{\frac{1}{2}}\|f\|_W \|x\|_2^{d-1}\|v\|_2.$$  

**Theorem (Kellogg’s Inequality)**

Let $F \in \{\mathbb{R}, \mathbb{C}\}$, $f \in \mathcal{H}^F_d[1]$, $x \in \mathbb{F}^{n+1}$ and $v \in \mathbb{F}^{n+1}$, then

$$|\overrightarrow{D}_xf \cdot v| \leq d\|f\|_{\infty} \|x\|_2^{d-1}\|v\|_2.$$  

Similar complexity analyses...
...with similar condition numbers

Complex setting:

$$\mu_{\text{norm}}(f, \zeta) := \|f\|_W \left\| D_\zeta f^\dagger \Delta^{1/2} \right\|_{2,2}.$$

$$\downarrow$$

$$M(f, \zeta) = \sqrt{q} \|f\|_\infty^C \left\| D_\zeta f^\dagger \Delta \right\|_{2,2}.$$
... with similar condition numbers

Complex setting:

\[ \mu_{\text{norm}}(f, \zeta) := \|f\|_W \left\| D_\zeta f^\dagger \Delta^{1/2} \right\|_{2,2}. \]

\[ \Downarrow \]

\[ M(f, \zeta) = \sqrt{q} \|f\|_\infty^C \left\| D_\zeta f^\dagger \Delta \right\|_{2,2}. \]

Real setting:

\[ \kappa(f) := \sup_{x \in \mathbb{S}^n} \frac{\|f\|_W}{\sqrt{\|f(x)\|_2^2 + \left\| D_x f^\dagger \Delta^{1/2} \right\|_{2,2}^{-2}}}. \]

\[ \Downarrow \]

\[ K(f) := \sup_{x \in \mathbb{S}^n} \frac{\sqrt{q} \|f\|_\infty^R}{\max \left\{ \|f(x)\|, \left\| D_x f^\dagger \Delta \right\|_{2,2}^{-1} \right\}}. \]
Any problems?

\[ \| \cdot \|_\infty \text{ is not cheap to estimate} \]

**Proposition**

Given \((f, k) \in \mathcal{H}_d^F[q] \times \mathbb{N}\) we can compute \(T\) such that

\[ (1 - 2^{-k})T \leq \|f\|_\infty \leq T \]

with cost

\[ \mathcal{O} \left( 2^{n \log n} D^n 2^{\frac{(k+1)n}{2}} N \right). \]

Gains are big enough to compensate for this
THREE Applications
1st Application:
Computing the Betti numbers of (Semi-)Algebraic Sets
State of the art

**Theorem**

There is a numerical algorithm \( \text{BETTI} \) that, given \( f \in \mathcal{H}_d[q] \), returns the Betti numbers of its zero set \( Z(f) \subset S^n \). The cost of \( \text{BETTI} \) on input \( f \) is bounded as

\[
\text{cost}(f) \leq 2^{O(n^2 \log n)} D^{O(n^2)} K(f)^{O(n^2)}.
\]

Furthermore, it satisfies

\[
\text{cost}(p) \leq q^{O(n)} (nD)^{O(n^3)}
\]

with probability at least \( 1 - (nqD)^{-n} \).

The result holds for a class of distributions extending the Gaussian

Outside a set of vanishingly small measure

this yields an exponential acceleration over all previous algorithms
The Algorithm
The Algorithm

\( \kappa(f) \) controls the mesh of the grid!
The Algorithm

\( \kappa(f) \) is in the criterion to determine which points are near!
The Algorithm

\[ \kappa(f) \] determines how big we should take the balls! (Through the Niyogi-Smale-Weinberger Theorem and a bound on the reach!)
The Algorithm

Union of Balls

some TDA (e.g. Nerve Lemma)

Betti numbers of zero set
(Even torsion coefficients!)
Replacing $\|\|_W$ with $\|\|_\infty$

(1) The same scheme can be applied using $K$ instead of $\kappa$

$$\frac{\text{cost}(\text{BETTI}_\infty, f)}{\text{cost}(\text{BETTI}_W, f)} \leq \left( \frac{K(f)}{\kappa(f)} \right)^{10n}$$

(2) For fixed $n$ and large $D$, the ratio in the right-hand side is of the order of

$$\left( \frac{C \sqrt{\ln(eD)}}{D^{n-\frac{1}{2}}} \right)^{10n}$$
2nd Application: The Plantinga-Vegter Algorithm
Given a real polynomial $f$, the PV algorithm meshes the real zero set.

Mostly used for two and three variables by computer graphics community, reported to be efficient, and quite popular.

Concretely speaking:
Given a polynomial $f \in \mathbb{R}[X, Y]$ (or $f \in \mathbb{R}[X, Y, Z]$) with degree $d$ it computes an isotopic piecewise linear approximation of the zero set of $f$ within a given square in $\mathbb{R}^2$ (cube in $\mathbb{R}^3$, respectively).

Ambiguous for precision control

Worst-case complexity analysis by Burr, Gao, Tsigaridas came after 14 years and predicted exponential running time

We use condition numbers for precision control and beyond-worst-case complexity analysis.
Smoothed Analysis of Algorithms

• Perturb a deterministic input $g$ with a random input $h$:

$$g + \sigma \|g\| h$$

where $\sigma \in (0, \infty)$ controls the “variance”

• For the algorithm of interest, we bound the quantity

$$\sup_g \mathbb{E}_h \text{cost}(g + \sigma \|g\| h)$$

- $\sigma = 0$ gives the worst-case complexity analysis
- $\sigma \to \infty$ gives the average case complexity analysis
- $\sigma \in (0, \infty)$ gives the smoothed complexity analysis

• Smoothed analysis explains run-time in practice!

• Note that we need to choose a probability distribution for $h$
In our case, $h$ is a dobro random polynomial, i.e., subgaussian coefficients with bounded continuous density
Worst-case case complexity of the PV algorithm

\[ 2^{\mathcal{O}(d^n)} \]

Smoothed complexity of the PV algorithm

With the Weyl norm,

\[ d^{\mathcal{O}(n^2)} \]

With the \( \infty \)-norm,

\[ (d \log d)^{\mathcal{O}(n)} \]

Smoothed complexity of the PV algorithm for low dimensions

<table>
<thead>
<tr>
<th></th>
<th>( n = 2 )</th>
<th>( n = 3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( PV_W )</td>
<td>( \mathcal{O}(d^8) )</td>
<td>( \mathcal{O}(d^{13}) )</td>
</tr>
<tr>
<td>( PV_\infty )</td>
<td>( \mathcal{O}(d^7 \log^{1.5}(d)) )</td>
<td>( \mathcal{O}(d^{10} \log^2(d)) )</td>
</tr>
</tbody>
</table>
3rd Application: Systems of complex quadratic equations
\[ f = q_1 \]
\[ g = q_0 \]
\[ q_t := t f + (1 - t) g \]
\( q_t := tf + (1 - t)g \)
\( d_S(q_i, q_{i+1}) := \frac{0.008535284}{\text{dist}_S(f, g) D^{3/2} \mu \text{norm}(q_i, z_i)^2} \)

\( z_{i+1} := N_{q_{i+1}}(z_i) \).
\[ d_S(q_i, q_{i+1}) := \frac{0.03}{\| f - g \|_\infty} \frac{\text{DM}(q_i, z_i)^2}{\| q_i \|_\infty} \]

\[ z_{i+1} := N_{q_{i+1}}(z_i). \]
<table>
<thead>
<tr>
<th>EXPECTED # STEPS</th>
<th>COST OF STEP</th>
<th>TOTAL COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W$</td>
<td>$\mathcal{O}(nD^{3/2}N)$</td>
<td>$\mathcal{O}(N)$</td>
</tr>
<tr>
<td>$\infty$</td>
<td>$\mathcal{O}(n^3D\log(eD))$</td>
<td>Large</td>
</tr>
</tbody>
</table>

The case of quadratic equations: $D = 2$ ($N = \mathcal{O}(n^3)$)

<table>
<thead>
<tr>
<th>EXPECTED # STEPS</th>
<th>COST OF STEP</th>
<th>TOTAL COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W$</td>
<td>$\mathcal{O}(n^4)$</td>
<td>$\mathcal{O}(n^3)$</td>
</tr>
<tr>
<td>$\infty$</td>
<td>$\mathcal{O}(n^3)$</td>
<td>$\mathcal{O}(n^{1.5+\omega})$</td>
</tr>
</tbody>
</table>

Note that $\omega < 2.375$!
Conclusion

As in the case of numerical linear algebra, a careful choice of norms can improve algorithm efficiency
¡Muchas Gracias!

Teşekkürler!

Eskerrik asko!