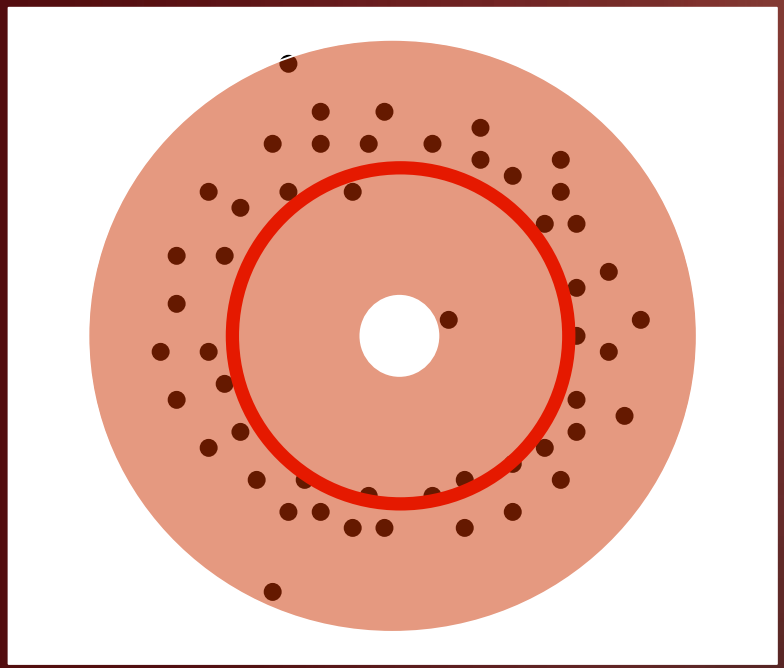
Shape Fitting With Outliers

Sariel Har-Peled and Yusu Wang
ACM SoCG 2003

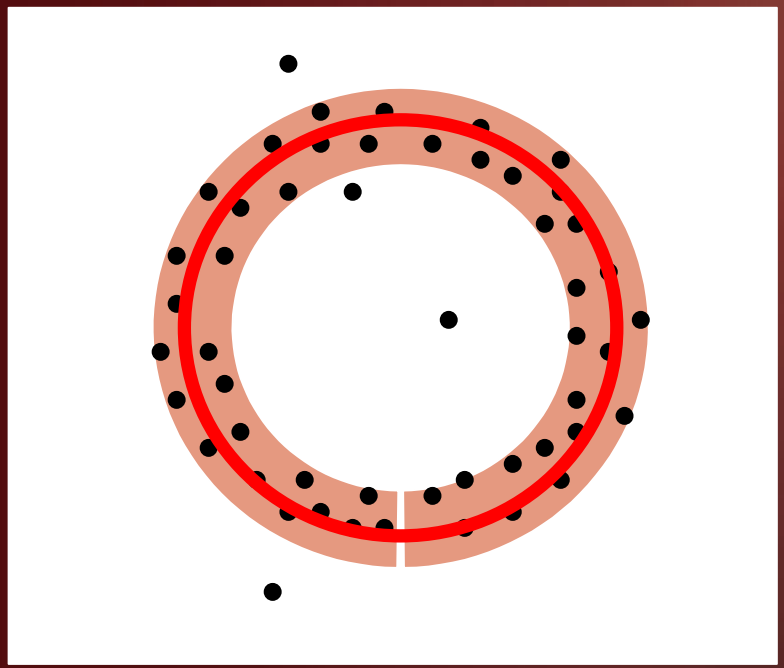
Paper Goal, I

- **Find the shape (of a given type) which approximates the best fit to a point set**
  - **Minimize the maximum distance from any point to the shape**
  - **Very sensitive to outliers**



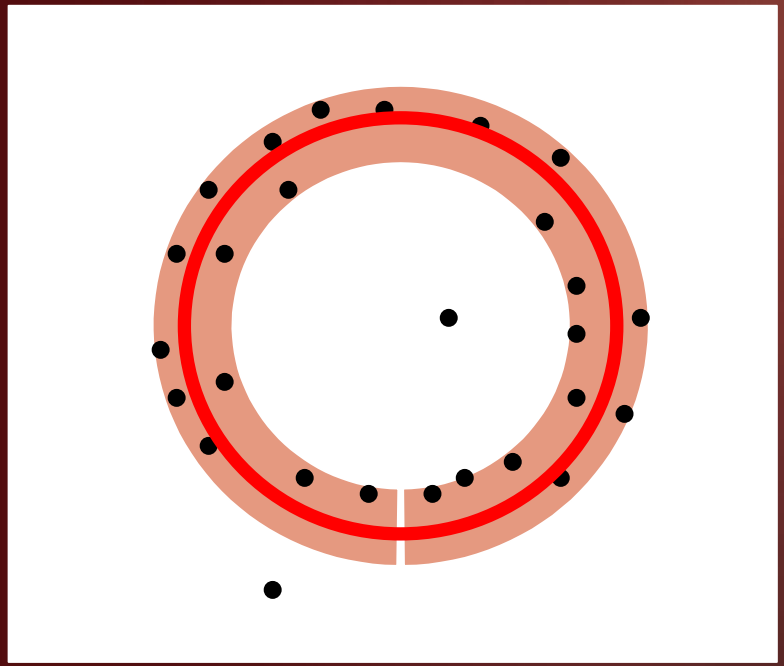
Paper Goal, II

- **Allow the fitting algorithm to ignore a fixed number, k, of outliers**



Paper Goal, III

- Find a coreset to allow efficient (O(n)) approximation
- Coreset construction algorithm is the paper's major contribution



k-Extent for Hyperplanes

- Shape fitting with outliers is related to this problem
- Given a set of n hyperplanes in $\Re^d$, find the shortest "vertical" line segment which intersects all but k of the hyperplanes

Coresets for the k-Extent of Hyperplanes

- Can be solved approximately using a coreset of the hyperplanes
- A good coreset for this problem can be used to find a good coreset of points for shape fitting

Definitions

For each of the definitions that follow, assume the following primitives:

- **x : Point in $\Re^{d-1}$**
- **A : Surface in $\Re^d$, $(x, f_A(x))$ over all x**
- **B : Surface in $\Re^d$, $(x, f_B(x))$ over all x**
- **H : Set of n hyperplanes in $\Re^d$**
- **ε : Constant > 0**
- **δ : Constant $\geq$ 0**

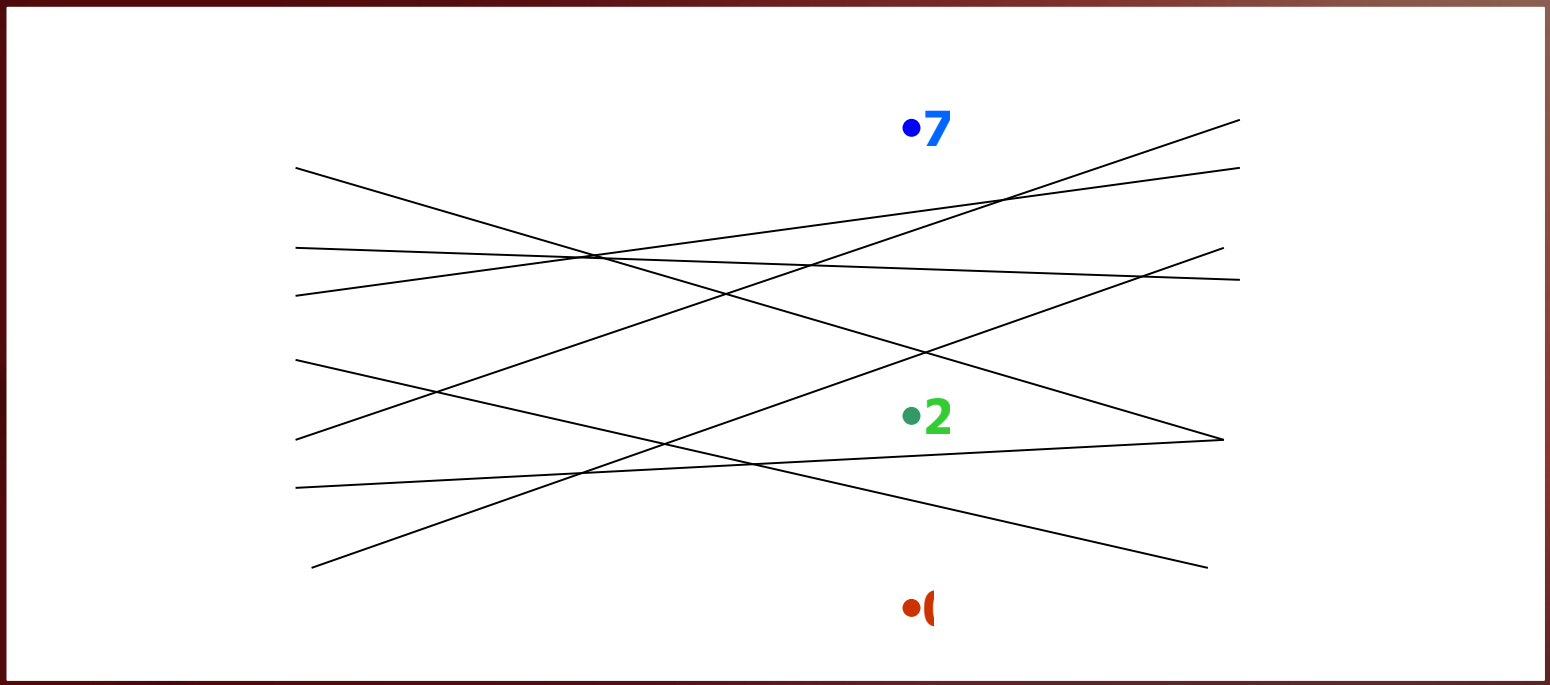
Definition: Vertical, Above/Below

- The **vertical** direction is parallel to the x_d-axis in $\Re^d$
- For any function f(x) defined for x in $\Re^{d-1}$, the surface (x,f(x)) in $\Re^d$ intersects any vertical line exactly once

- A is **above** B at x if $f_A(x) \geq f_B(x)$
- A is **below** B at x if $f_A(x) \leq f_B(x)$

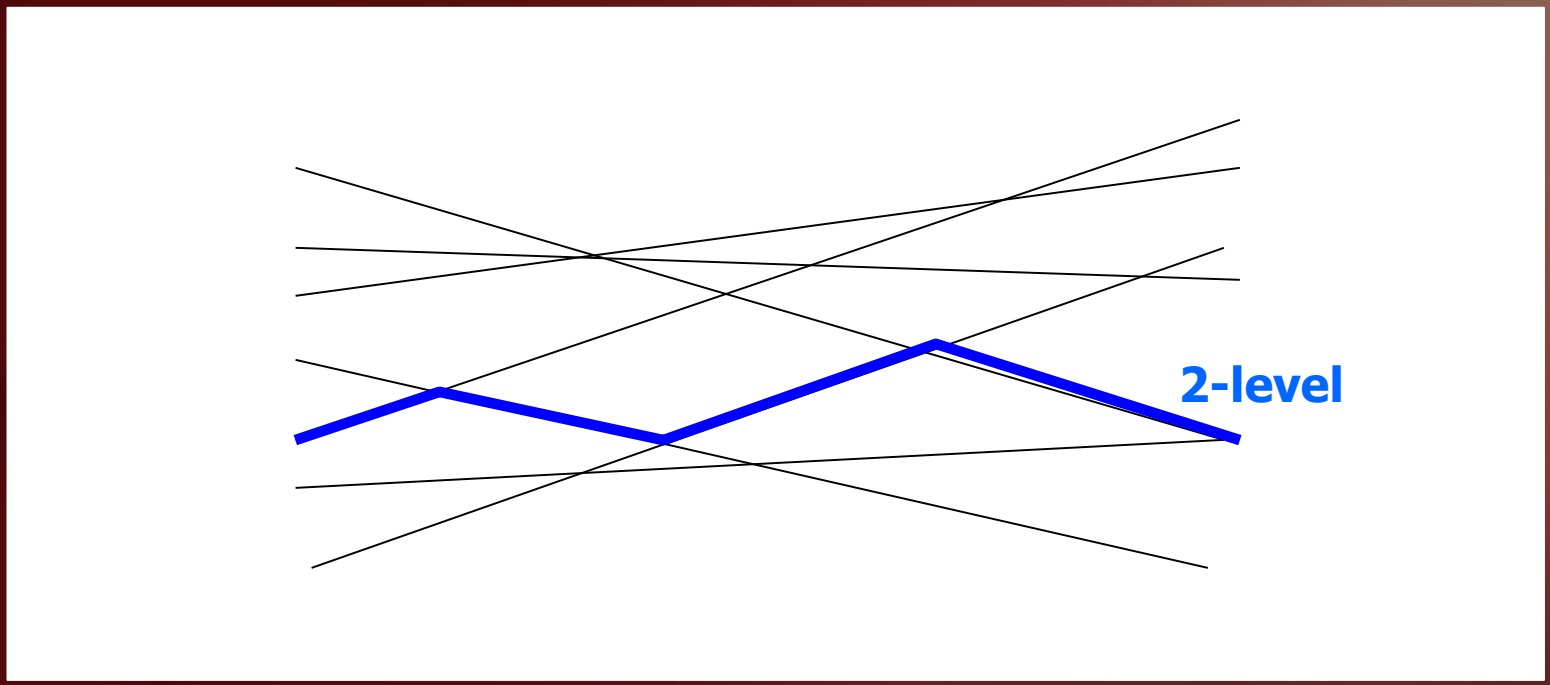
Definition: Level

- The **level** of a point (x,f(x)) relative to H is the number of hyperplanes in H below that point



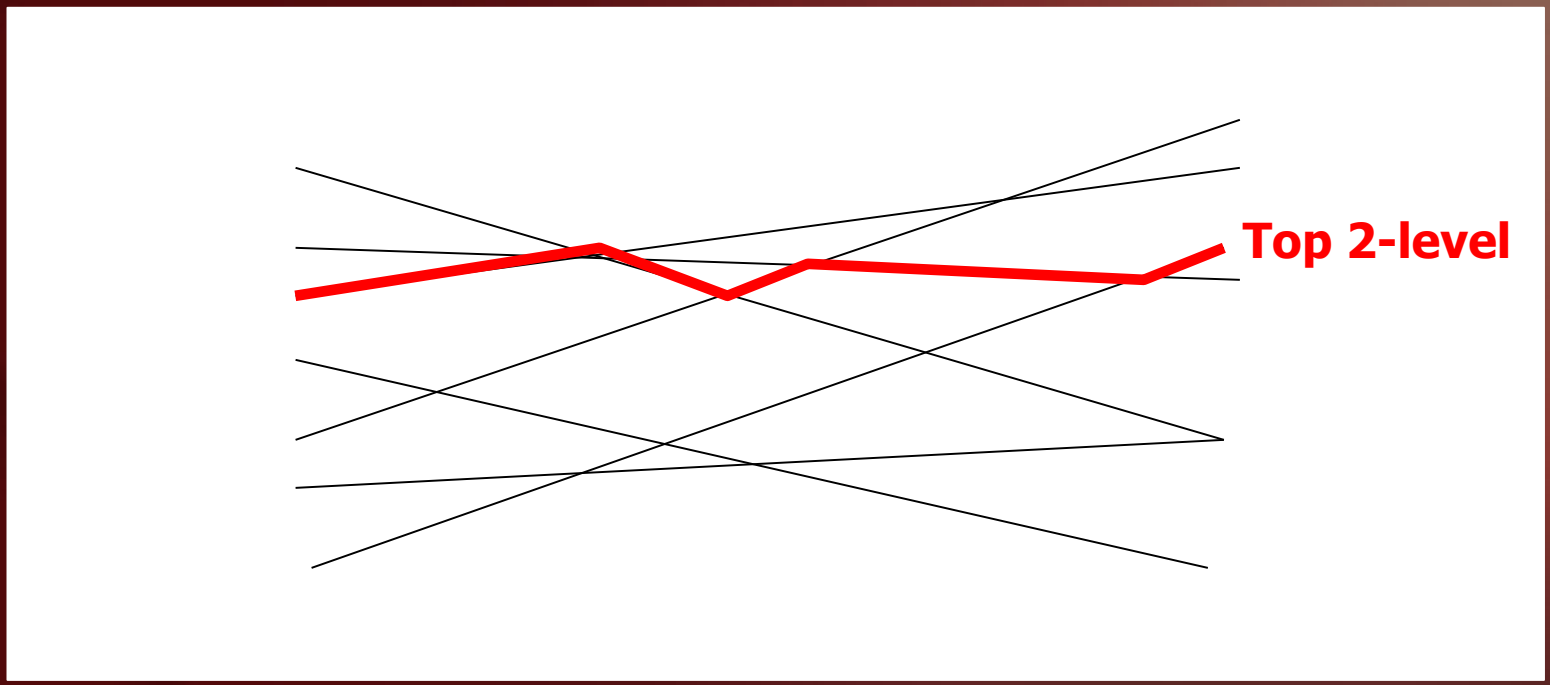
Definition: k-Level

- The **k-level** of H (written $L_{H,k}$) is the closure of all points in H whose level is k
- $L_{H,k}(x) = x_d$, such that (x,x_d) is in $L_{H,k}$



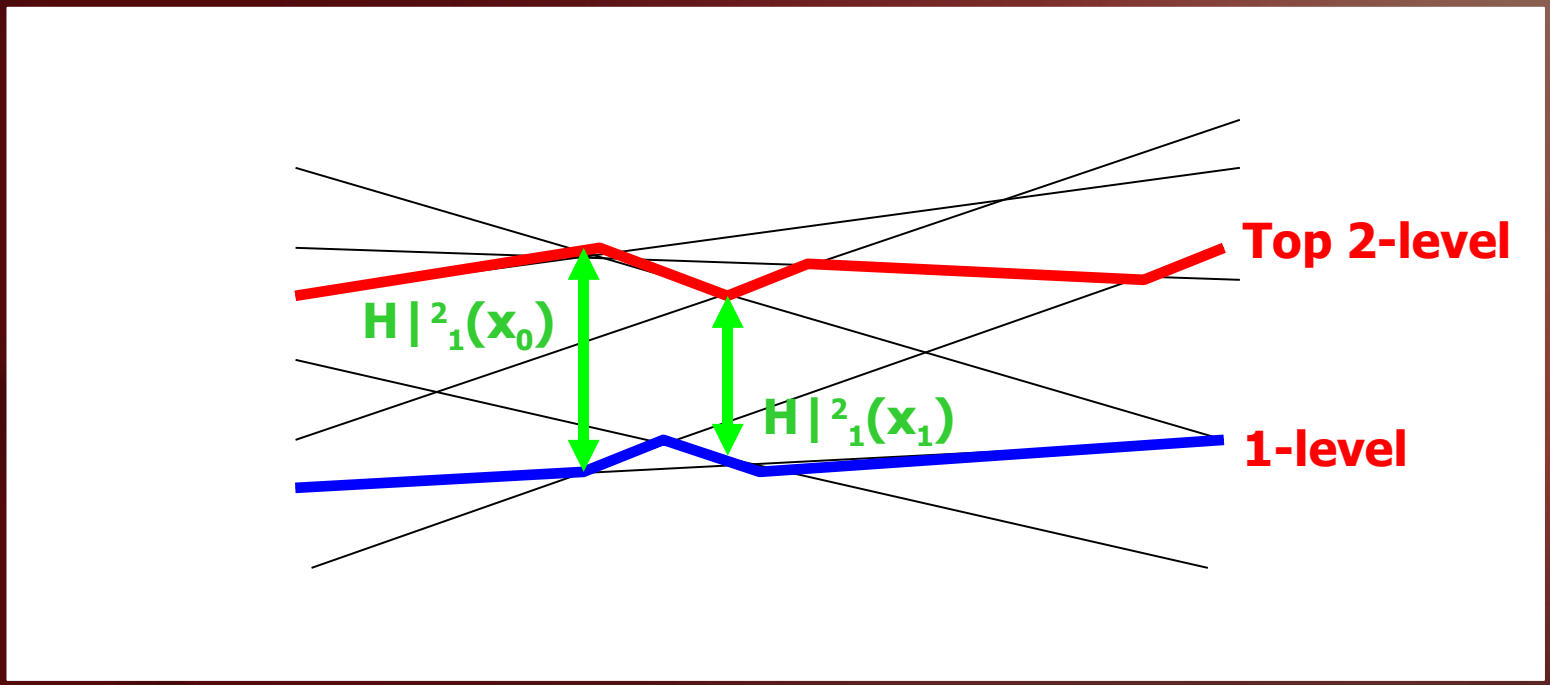
Definition: Top k-Level

- The **top k-level** of H (written $U_{H,k}$) is the closure of all points in H whose level is n-k-1
- $U_{H,k}(x) = x_d$, such that (x,x_d) is in $U_{H,k}$



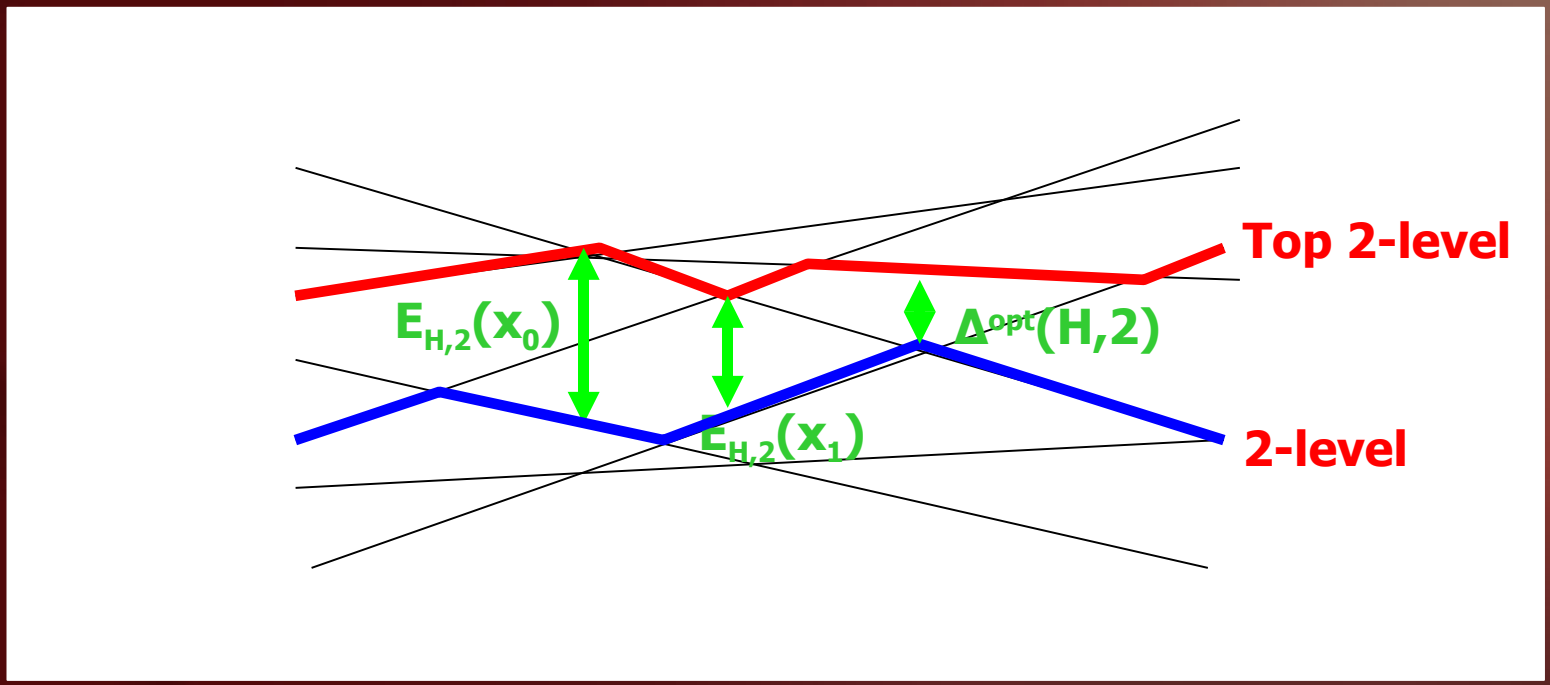
Definition: (k,r)-extent

- The *(k,r)-extent* of H at x (written $H|^k_r(x)$) is the vertical distance between the r-level and the top k-level of H



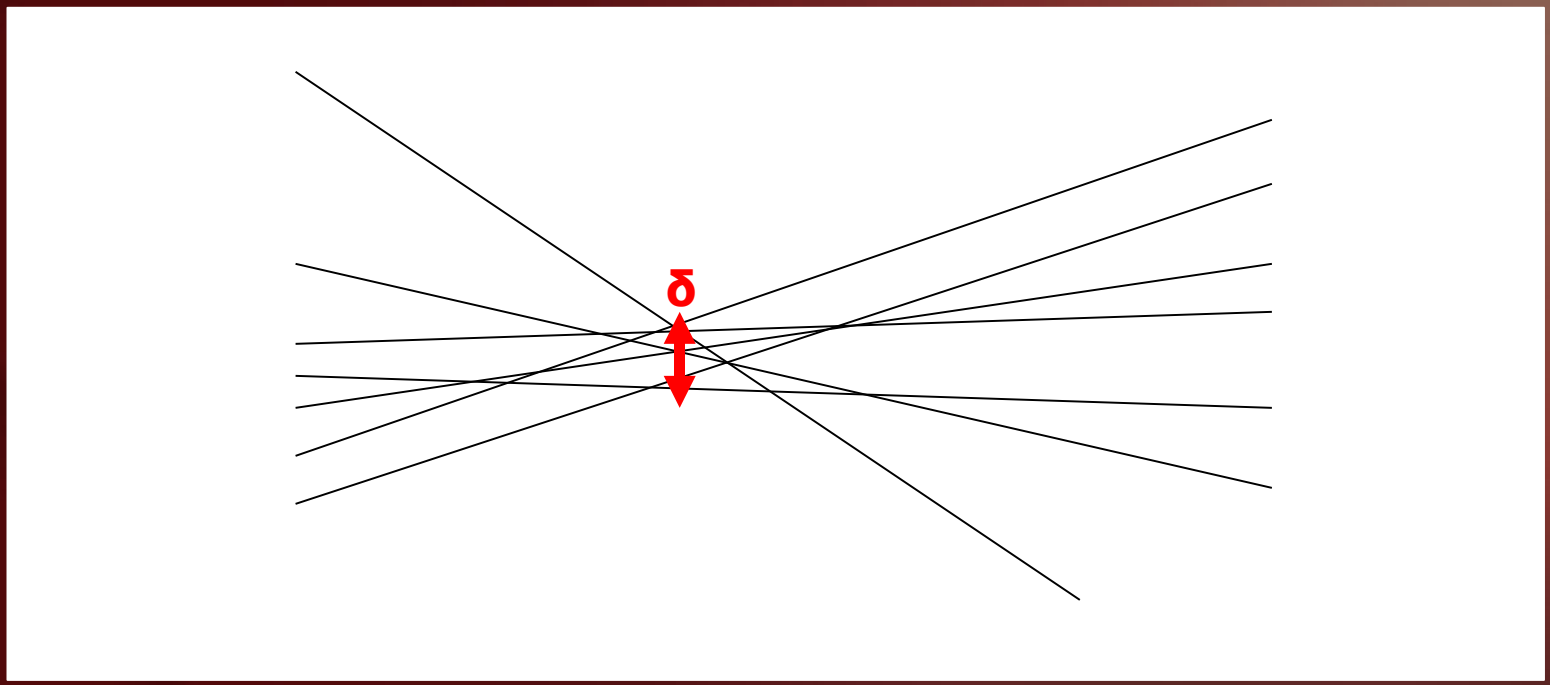
Definition: k-extent

- The **k-extent** of H at x (written $E_{H,k}(x)$) is shorthand for the (k,k)-extent
- Notation $\Delta^{opt}(H,k)$ refers to the smallest k-extent over all x



Definition: δ-sheaf

- A set of hyperplanes, J, is a δ-sheaf if $\Delta^{opt}(J,0) \leq \delta$.
- A 0-sheaf is simply a sheaf.



Definition: ε-approximation to $H|^k_r$

- An **ε-approximation to $H|^k_r$** is a function $E_\varepsilon : \Re^{d-1} \to \Re$ such that

$$(1-\varepsilon)H|^k_r(x) \leq E_\varepsilon(x) \leq H|^k_r(x)$$

Definition: (k,ε,δ)-coreset

- A **(k,ε,δ)-coreset** of H at x is a subset H′ of H such that for any r<k,

$$L_{H,r}(x) \le L_{H',r}(x) \le L_{H,r}(x) + \varepsilon E_{H,r}(x) + \delta$$

$$U_{H,r}(x) - \varepsilon E_{H,r}(x) - \delta \le U_{H',r}(x) \le U_{H,r}(x)$$

- A **(k,ε)-coreset** is one with δ=0

Overview: Computing (k,ε)-Coresets for Sets of Hyperplanes

- Start with hyperplane set H
- Select O(1) vertical segments of bounded length which collectively intersect all but O(k) hyperplanes in H
- Use segments to subdivide H into O(1/ε) disjoint subsets
- Compute $O(k/\varepsilon^{d-2})$-sized coreset for each subset
- Merge subset coresets to get a coreset of size $O(k/\varepsilon^{d-1})$ for H

Lemma 3.1

Let J be a set of n hyperplanes in $\Re^d$.

In O(n) time, one can compute a set S of O(1) vertical segments, such that with high probability:

- All segments are no longer than $\Delta^{opt}(J,k)$
- O(k) hyperplanes are not stabbed by S

Lemma 3.1 Proof: Algorithm

- Randomized, iterative algorithm
- At step i of the algorithm, let:
  - S_i = Set of segments created
  - Q_i = Set of hyperplanes not intersected by S_i
- S_1 = empty set, Q_1 = J

Lemma 3.1 Proof: Algorithm

- Case 1 : If $|Q_i| = O(k)$
  - then done
- Case 2 : If $|Q_i| \leq n^{1/(3d)}$
  - compute k-level and top k-level
  - find shortest vertical segment connecting the two levels
  - add that segment to S_{i+1}
  - computable in $O(|Q_i|^{2d+1}) = O(n)$ time
- Otherwise…

Lemma 3.1 Proof: Algorithm

- Case 3 : If $|Q_i| \geq n^{1/(3d)}$
  - pick random sample, R_i, from Q_i of size $O(n^{1/(3d)}\log n)$
  - compute the K_i-level and K_i top-level of R_i, where $K_i = \lceil |R_i|(k/|Q_i| + \tau) \rceil$ and $\tau = 1/n^{1/6d}$
  - find the shortest segment connecting the two levels
  - add that segment to S_{i+1}
  - computable in $O(|Q_i|^{2d+1}) = O(n)$ time

Lemma 3.1 Proof : Bounding the Number of Case 3 Iterations

- The number of distinct hyperplane sets that can be crossed by any vertical segment is bounded by $O(n^{d+1})$
- The range space has a bounded Vapnik-Chervonenkis dimension, so the ε-sample theorem holds:

$$\frac{|Q_i \cap s|}{|Q_i|} - \tau \leq \frac{|R_i \cap s|}{|R_i|} \leq \frac{|Q_i \cap s|}{|Q_i|} + \tau,$$

- (for any vertical segment s, and $\tau = 1/n^{1/6d}$)

Lemma 3.1 Proof : Bounding the Number of Case 3 Iterations

- With high probability
  - segment created is shorter than $\Delta^{opt}(J,k)$
  - intersects at least $|Q_i|(1-4\tau)-2k$ hyperplanes of Q_i, where $\tau = 1/n^{1/6d}$
  - $|Q_{i+1}| \leq 4\tau|Q_i| + 2k$

- For $|Q_i| \geq k/\tau$

  $|Q_{i+1}| \leq 6\tau|Q_i| \leq (6/n^{1/6d})|Q_i|$
- For $|Q_i| \leq k/\tau$

  $|Q_{i+1}| \leq 6k$ (final iteration)

- Total number of iterations : $O(\log_{1/\tau} n)=O(6d)=O(1)$

Forming Disjoint Subsets of H

- Create set of O(1) vertical segments, S
- Divide each segments S into $\lceil 4/\varepsilon \rceil$ equal subsegments
- Subsegment lengths $\leq \varepsilon\Delta^{opt}(H,k)/4$
- Associate each hyperplane of H that intersects S with exactly one of its intersecting subsegments
- Each of the $O(1/\varepsilon)$ subsets is a $(\varepsilon\Delta^{opt}(H,k)/4)$-sheaf

Lemma 3.4

- Let J and J′ be hyperplane sets
- Let $J_0,\ldots,J_m$ be a partition of J
- Let $J'_0,\ldots,J'_m$ be a partition of J′

- If J'_i is a (k,ε,δ)-coreset for J_i for $i=0,\ldots,m$, then J′ is a (k,ε,δ)-coreset for J

Lemma 3.4 Proof

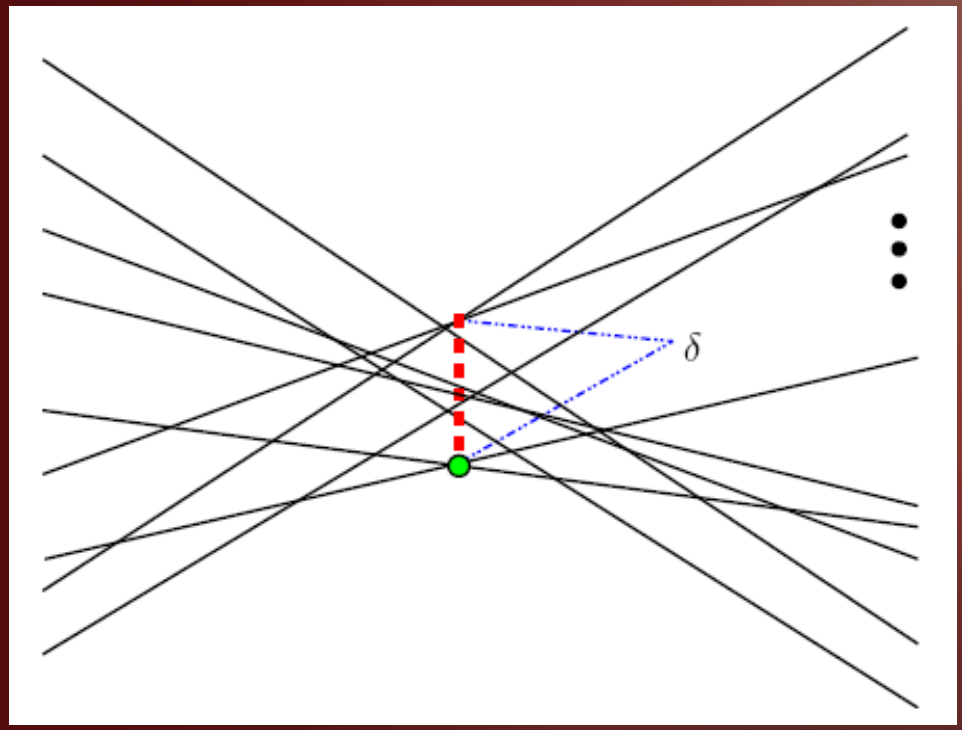
- At any point x in $\Re^{d-1}$, let $i_0,\ldots,i_m$ be the number of hyperplanes in $J_0,\ldots,J_m$ below $(x,L_{J,r}(x))$
- Note that $L_{J,r}(x) = \max_j L_{J_j,i_j}(x)$

$$
\begin{aligned}
\mathbf{L}_{\mathcal{J},r}(x) &\leq \mathbf{L}_{\mathcal{J}',r}(x) \leq \max_{j=1}^{m} \mathbf{L}_{\mathcal{J}'_j,i_j}(x) \\
&\leq \max_{j=1}^{m} \left(\mathbf{L}_{\mathcal{J}_j,i_j}(x) + \varepsilon \mathcal{E}_{\mathcal{J}_j,k}(x) + \delta\right) \\
&\leq \left(\max_{j=1}^{m} \mathbf{L}_{\mathcal{J}_j,i_j}(x)\right) + \varepsilon \mathcal{E}_{\mathcal{J},k}(x) + \delta \\
&= \mathbf{L}_{\mathcal{J},r}(x) + \varepsilon \mathcal{E}_{\mathcal{J},k}(x) + \delta.
\end{aligned}
$$

- Analogous argument bounds $U_{J',r}(x)$

Coresets for the 2D Case, I

- Each of the subsets of H is a 2D δ-sheaf
- Select coreset of O(k) lines which approximately preserve the first k levels and top levels



Lemma 3.5

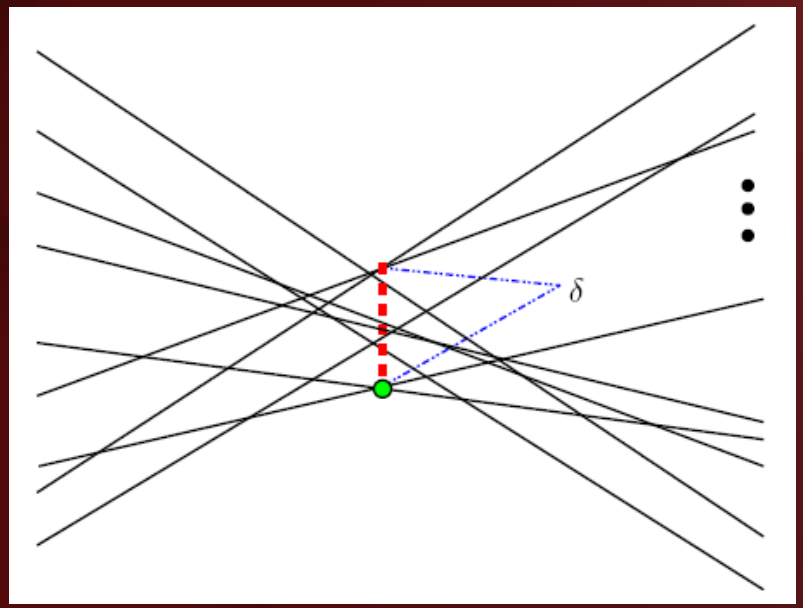
- **Given:**
  - J : Set of hyperplanes in $\Re^d$
  - $\underline{J}$: Hyperplanes of J, each independently shifted down by some distance $0 \le \delta_h \le \delta$

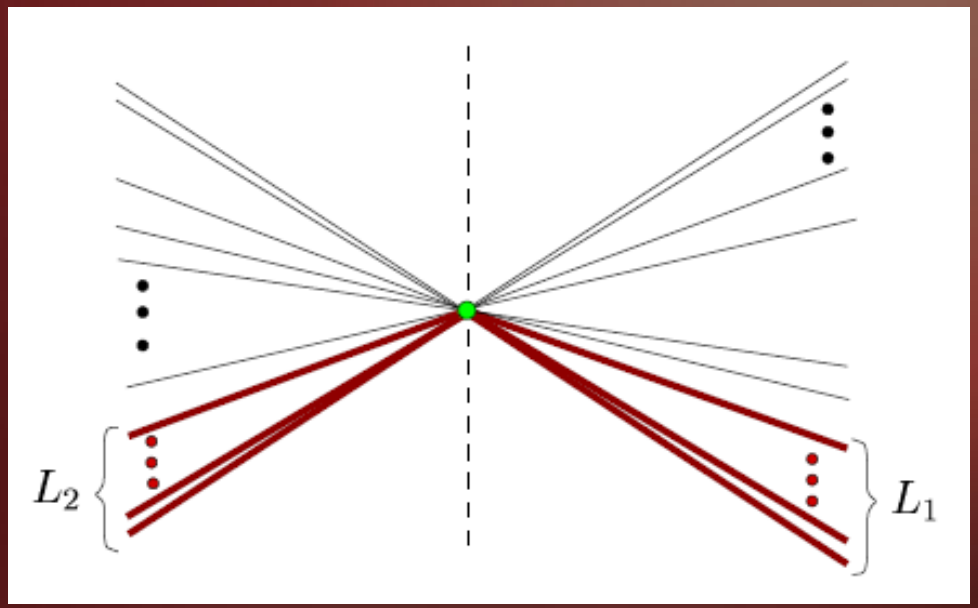
- **For any x in $\Re^{d-1}$ and $1 \le r,k \le n$:**
  - $L_{J,r}(x)-\delta \le L_{\underline{J},r}(x) \le L_{J,r}(x)$
  - $U_{J,r}(x)-\delta \le U_{\underline{J},r}(x) \le U_{J,r}(x)$
  - $E_{J,k}(x)-\delta \le E_{\underline{J},k}(x) \le E_{J,k}(x)+\delta$

- **$\underline{J}$ is a $(k,0,\delta)$-coreset of J**

Coresets for the 2D Case, II

- Consider sheaf with lines parallel to δ-sheaf's
- Keep only k lines with max and min slope
- Shift kept lines up by ≤δ to return to normal positions
- Result is (k,0,δ)-coreset to δ-sheaf





Observation 3.6

- **Given:**
  - **H : set of hyperplanes in $\Re^d$**
  - **H′ : (k,ε,δ)-coreset for H**
- **H′ is a (k,μ)-coreset for H, where $\mu=\varepsilon+\delta/\Delta^{opt}(H,k)$**

Coresets for the 2D Case, III

- Partition creates δ-sheafs, where

  $\delta = (\varepsilon \Delta^{opt}(H,k)/4)$

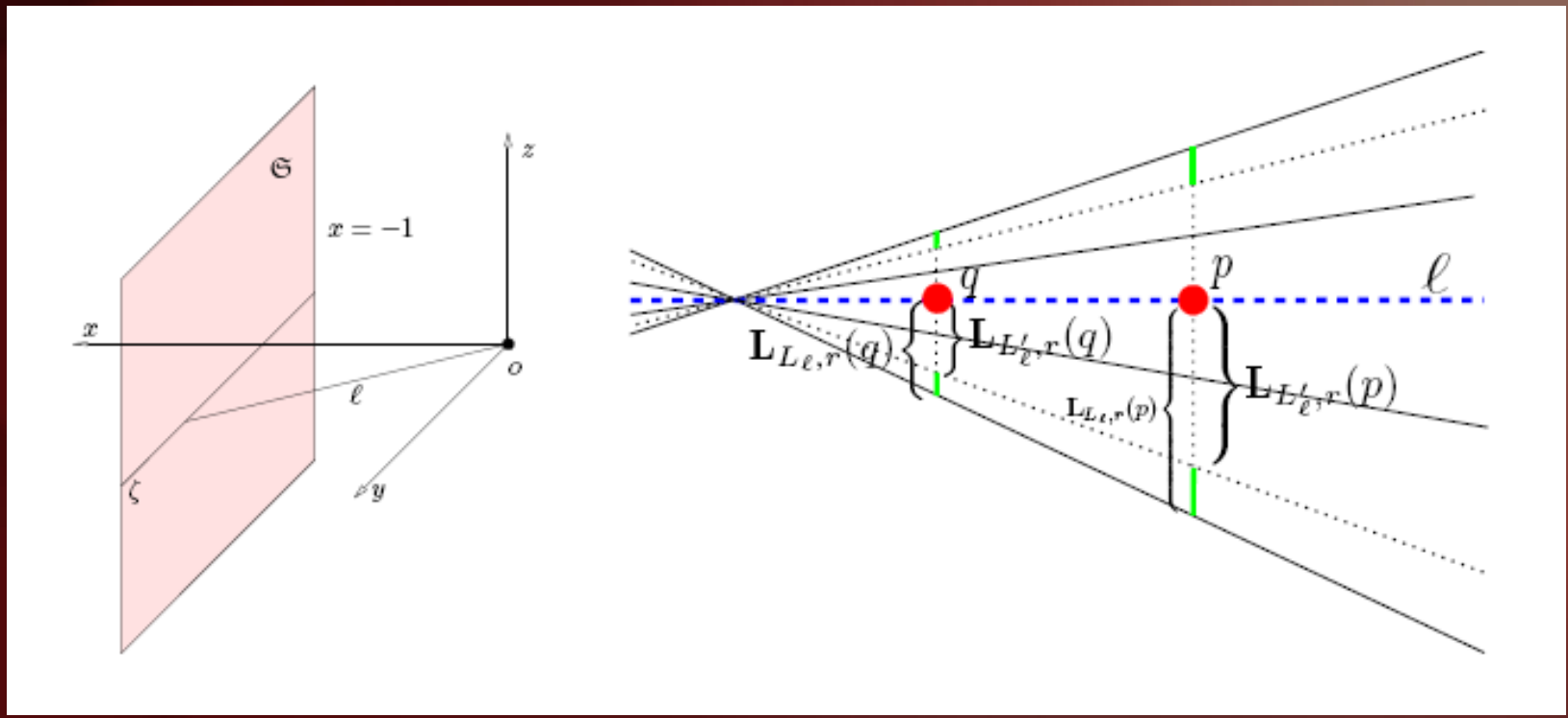
- Create (k,0,δ)-coresets to δ-sheafs

- By observation 3.6, coresets are (k,ε)-coresets, since

  $\mu = 0+\delta/\Delta^{opt}(H,k) = \varepsilon/4$

Coresets for the 3D Case

- Algorithm uses result from 2D case
- Computes coresets of size O(k/ε) in O(n+k/ε) time



Theorem 3.13: Coresets for Higher Dimensions

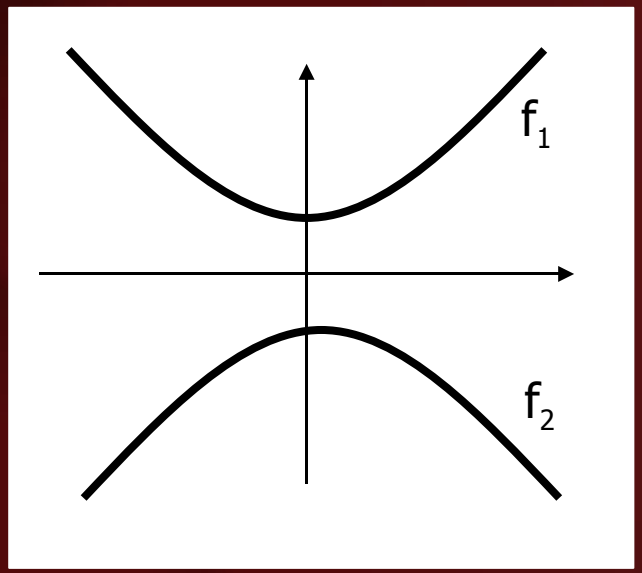
- **Given:**
  - **H : Set of n hyperplanes in $\Re^d$**
  - **k, $\varepsilon > 0$**
- **With high probability, one can compute a subset H′ of H such that:**
  - **H′ is a (k,ε)-coreset of H**
  - **$|H'| = O(k/\varepsilon^{d-1})$**
- **Requires $O(n+k/\varepsilon^{d-1})$ time**

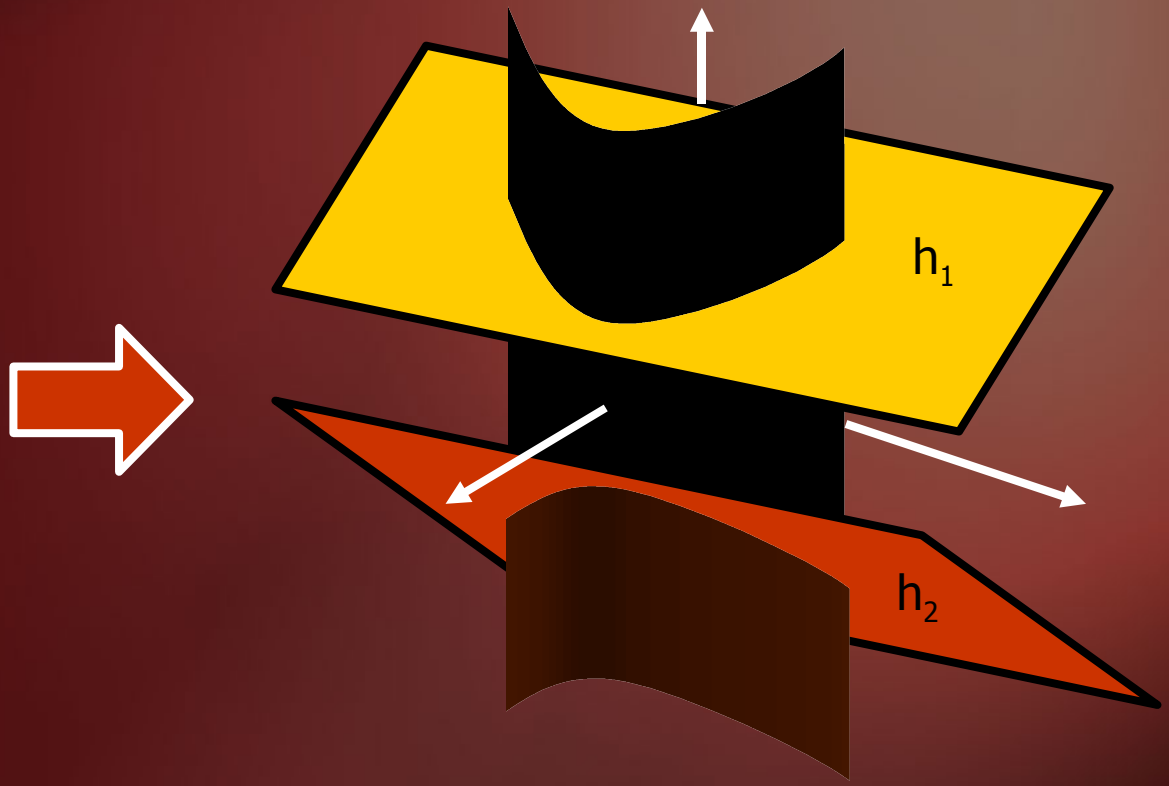
Linearization of Multivariate Polynomials

- Let $F=\{f_1,\ldots,f_n\}$ be polynomials in $\Re^d$
- Linearizing the polynomials
  - Replace each monomial in F with a new, distinct variable, to create hyperplane set H
  - Variable remappings define a "linearization image" from old variables to new ones
- For example:
  - $f_i(x)=a_ix^2+b_ix+c_i \Rightarrow h_i(u,v)=a_iu+b_iv+c_i$
  - Linearization Image : $\eta(x) = (u,v) = (x,x^2)$
  - H corresponds to F only where $v=u^2$

Coresets for Polynomials

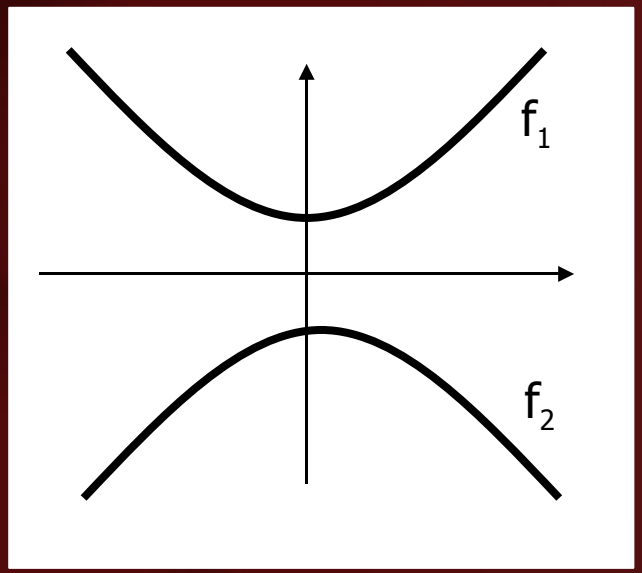
- H only accurately represents F within a (nonconvex) subset of points, X
- H is not directly useful for finding the minimum extent of F

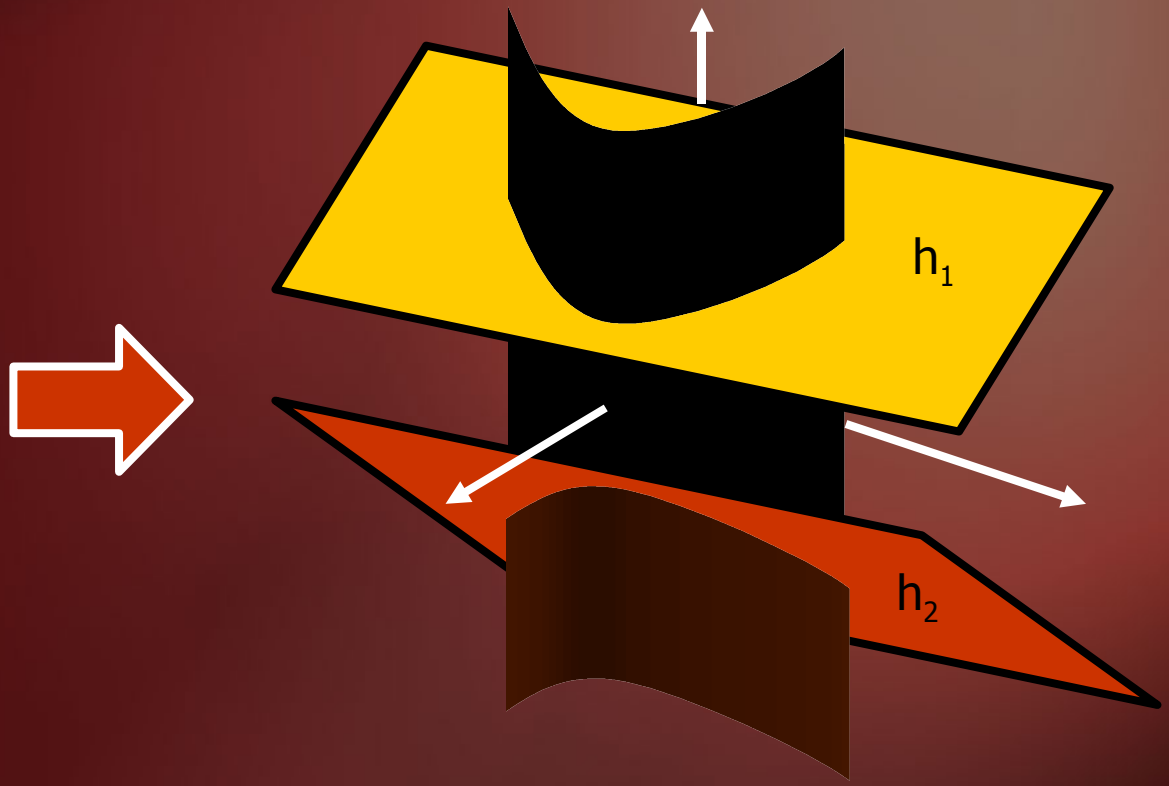




Coresets for Polynomials

- Within X, levels and extent of F is faithfully represented in H
- Coresets for H imply coresets for F





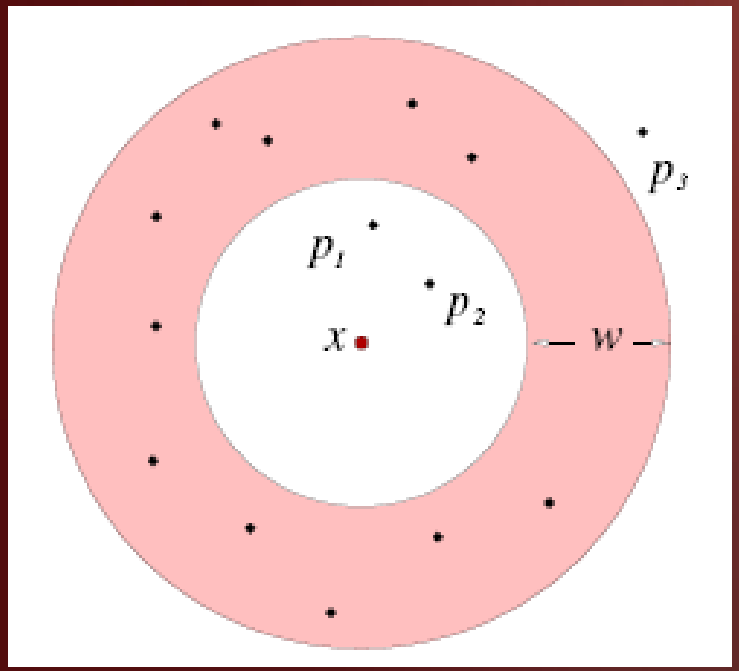
Theorem 4.4: Coresets for Square Roots of Polynomials

- **Given:**
  - $F=\{f_1^{1/2},\ldots,f_n^{1/2}\}$ (f_i 's are polynomials on $\Re^d$)
  - $0 \leq \varepsilon \leq 1$
- **Let s be the number of distinct monomials present in F**

- **Can compute subset F′ of F such that:**
  - F′ is (k,ε)-sensitive for F
  - $|F'| = O(k/\varepsilon^{2s})$
- **Requires $O(n+k/\varepsilon^{2s})$ time**

Application: Min-Width Annulus

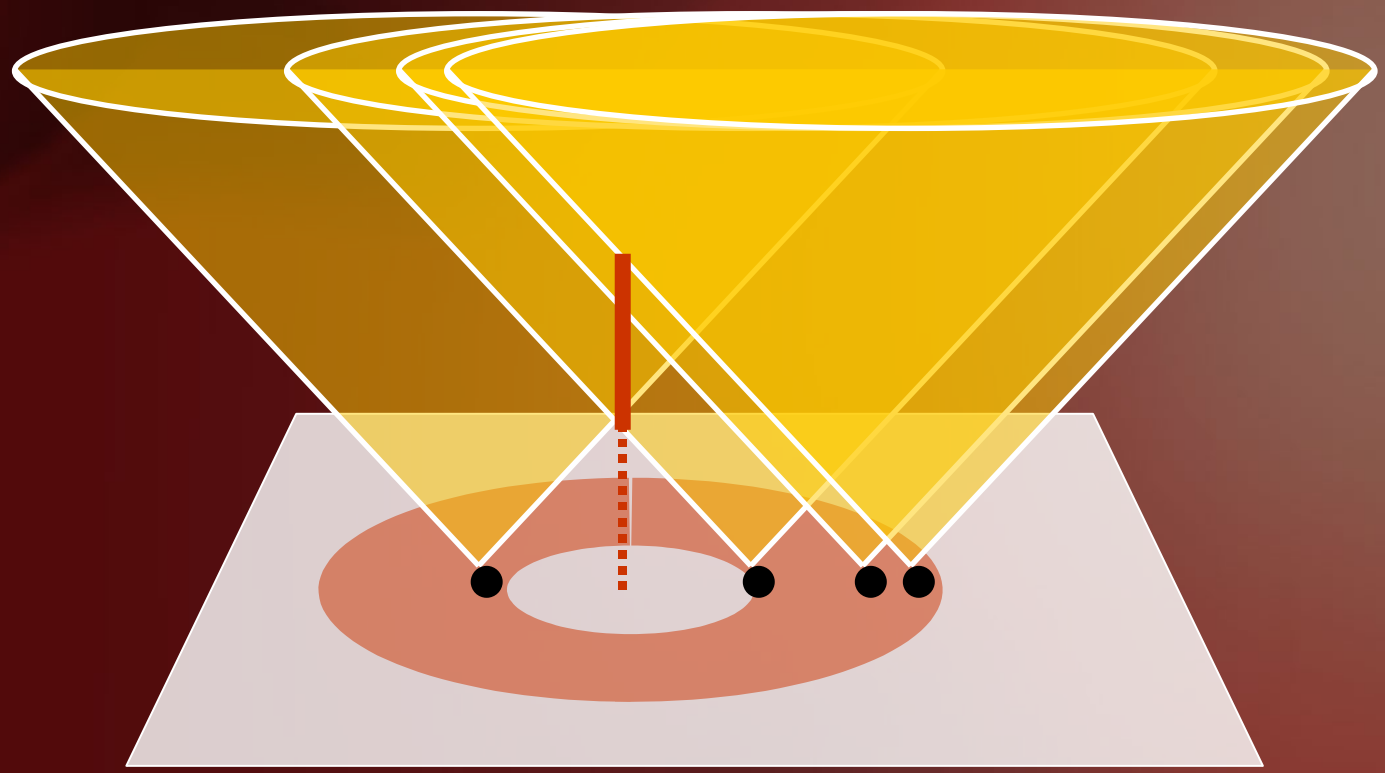
- Given a set of points in $\Re^d$
- Find the thinnest spherical shell that contains the points



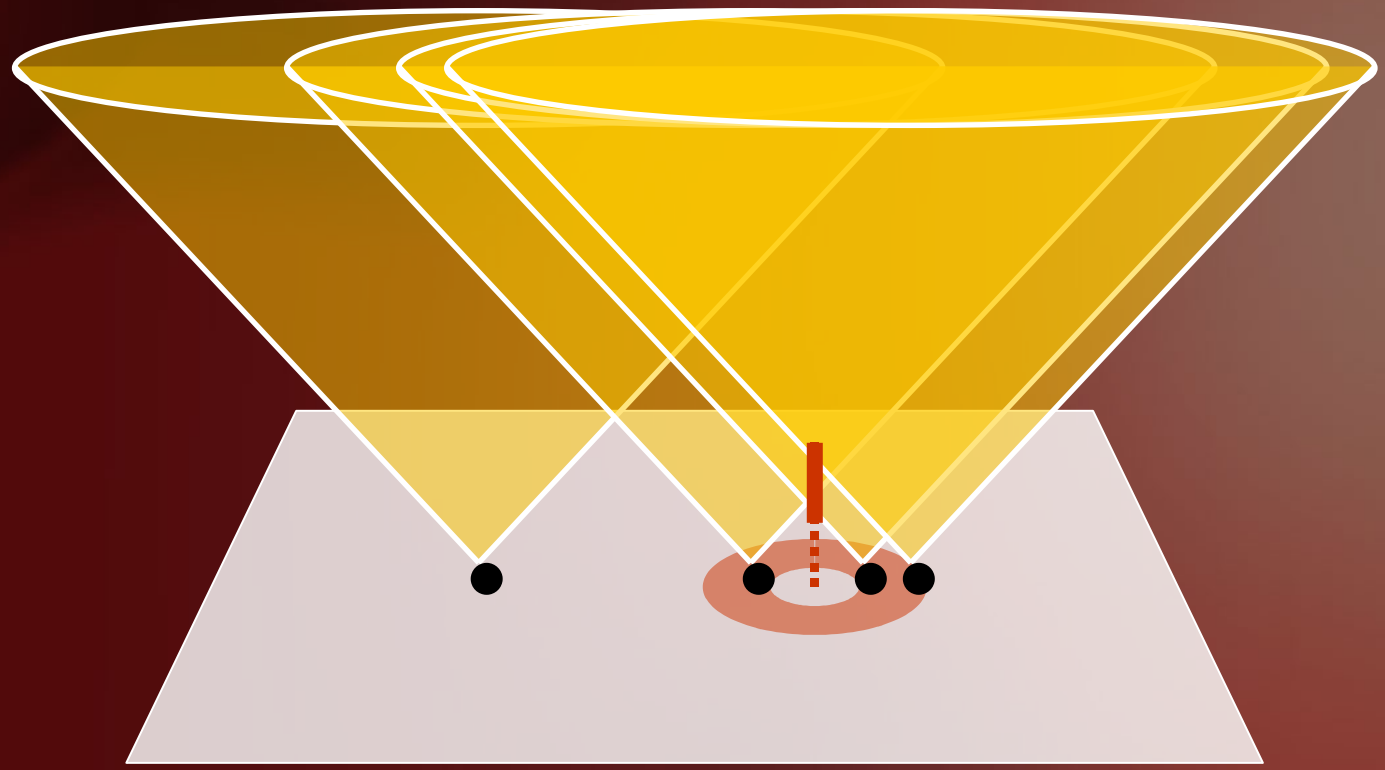
Approach: Min-Width Annulus

- Parametrize over annulus center, c, in $\Re^d$
- Squared distance from c to p_i defines a polynomial function $f_i(c)$ over $\Re^d$
- Define $F = \{f_1^{½}, \ldots, f_{|P|}^{½}\}$
- Optimal center occurs at the point of min extent of F

Application: Min-Width Annulus



Application: Min-Width Annulus



Approach: Min-Width Annulus

- Define $F = \{f_1^{½}, \ldots, f_{|P|}^{½}\}$
- Compute a (k,ε)-coreset, F', for F
- Find the point of min extent for F' by brute force
- Requires $O(n + k^{2d+1}/\varepsilon^{4d(d+1)})$ time

Conclusion

- For many shape-fitting problems, we can build coresets whose size depends only on ε and k
- Can create coresets in O(n) time
- Allows shape-fitting complexity to be independent of n
- Limitations
  - k must be very small ($O(n^{1/2d})$)
  - k must be specified before processing