Utrecht University

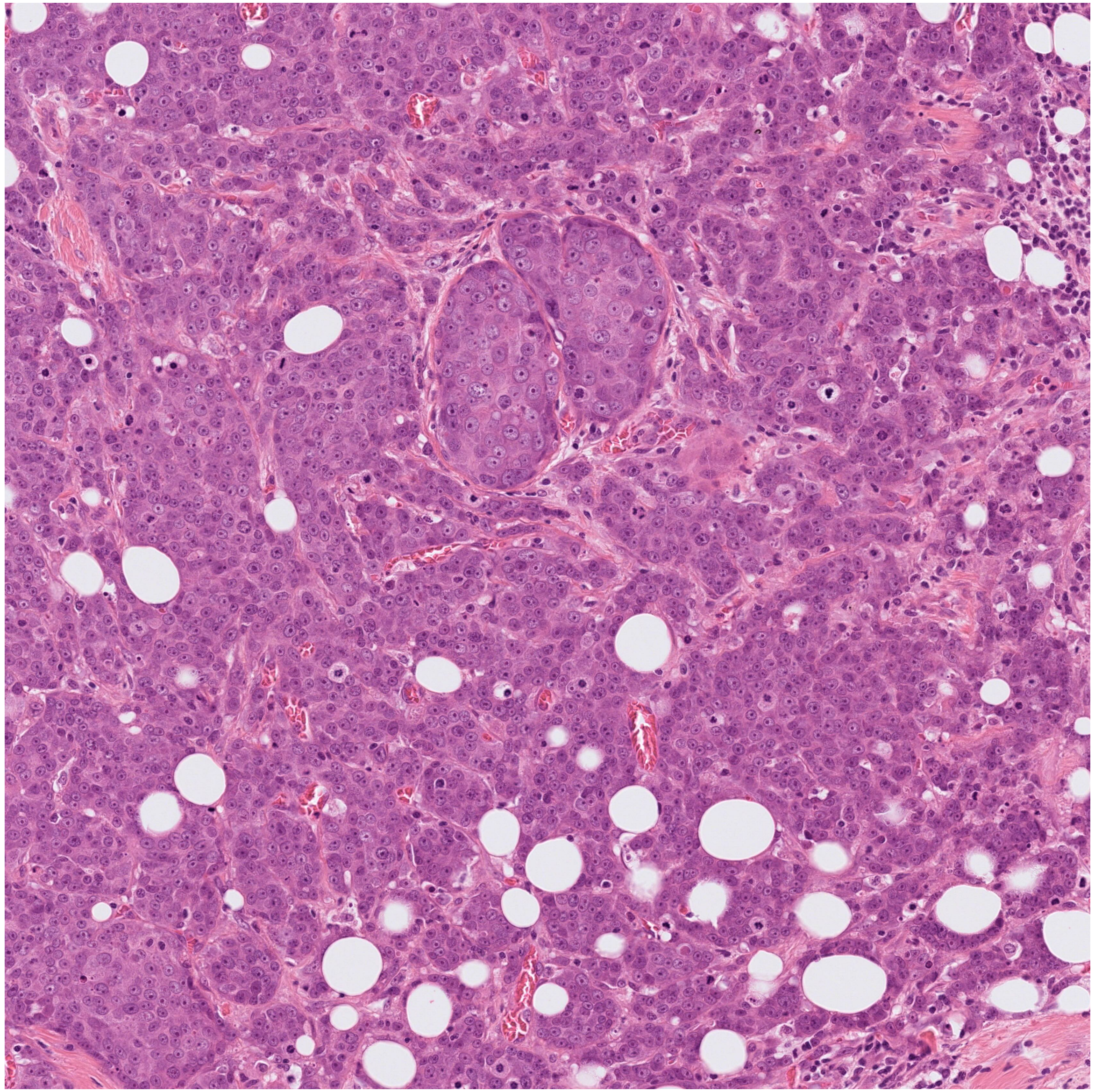
Computing Largest Subsets of Points Whose Convex Hulls have Bounded Area and Diameter

Gianmarco Picarella [1] Marc van Kreveld [1] Frank Staals [1] Sjoerd de Vries [1, 2]

[1] Department of Information and Computing Science, Utrecht University, The Netherlands
[2] Department of Digital Health, University Medical Center Utrecht, The Netherlands

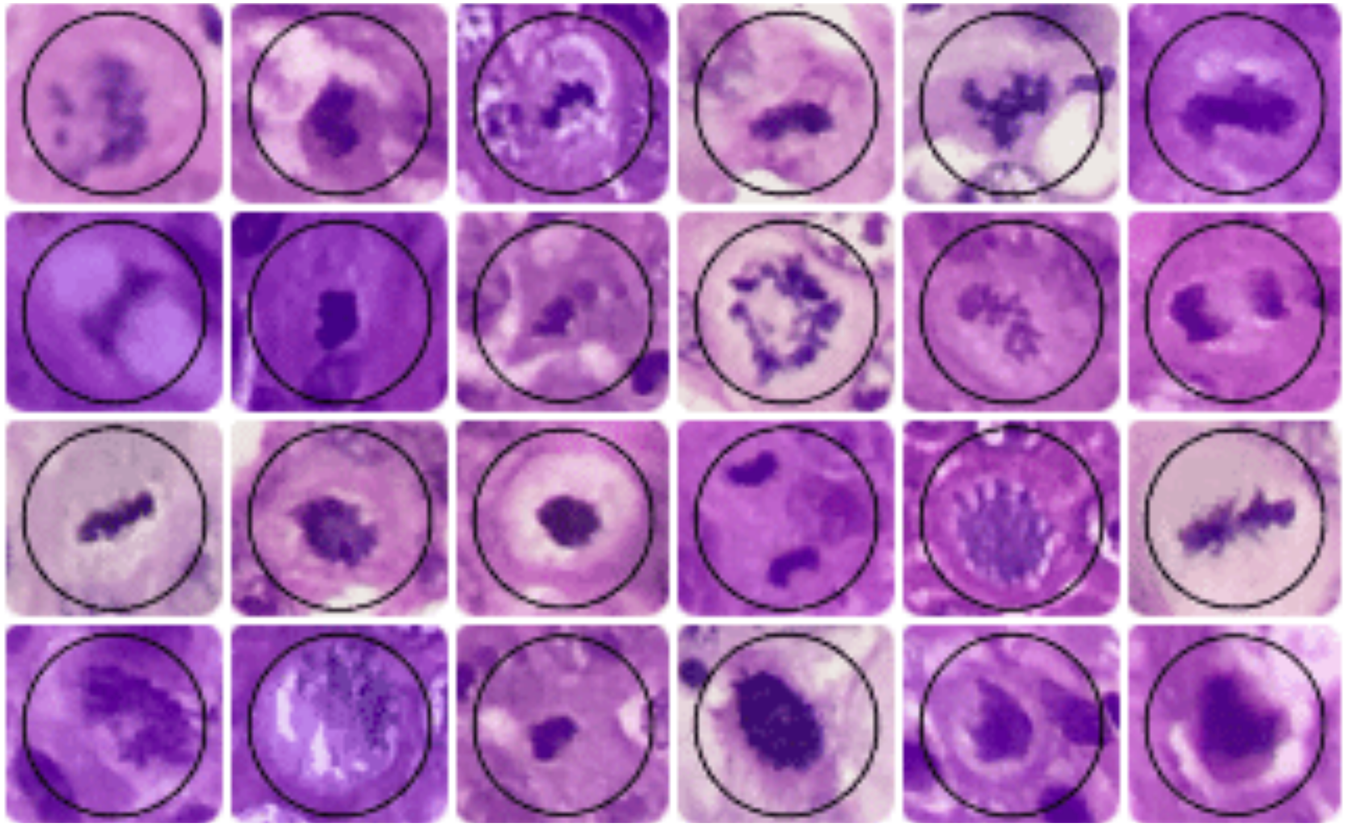
Motivation

- Medical images are examined by pathologists to detect anomalies
- Whole-Slide Images (WSI) are digitalized histological images of human tissues



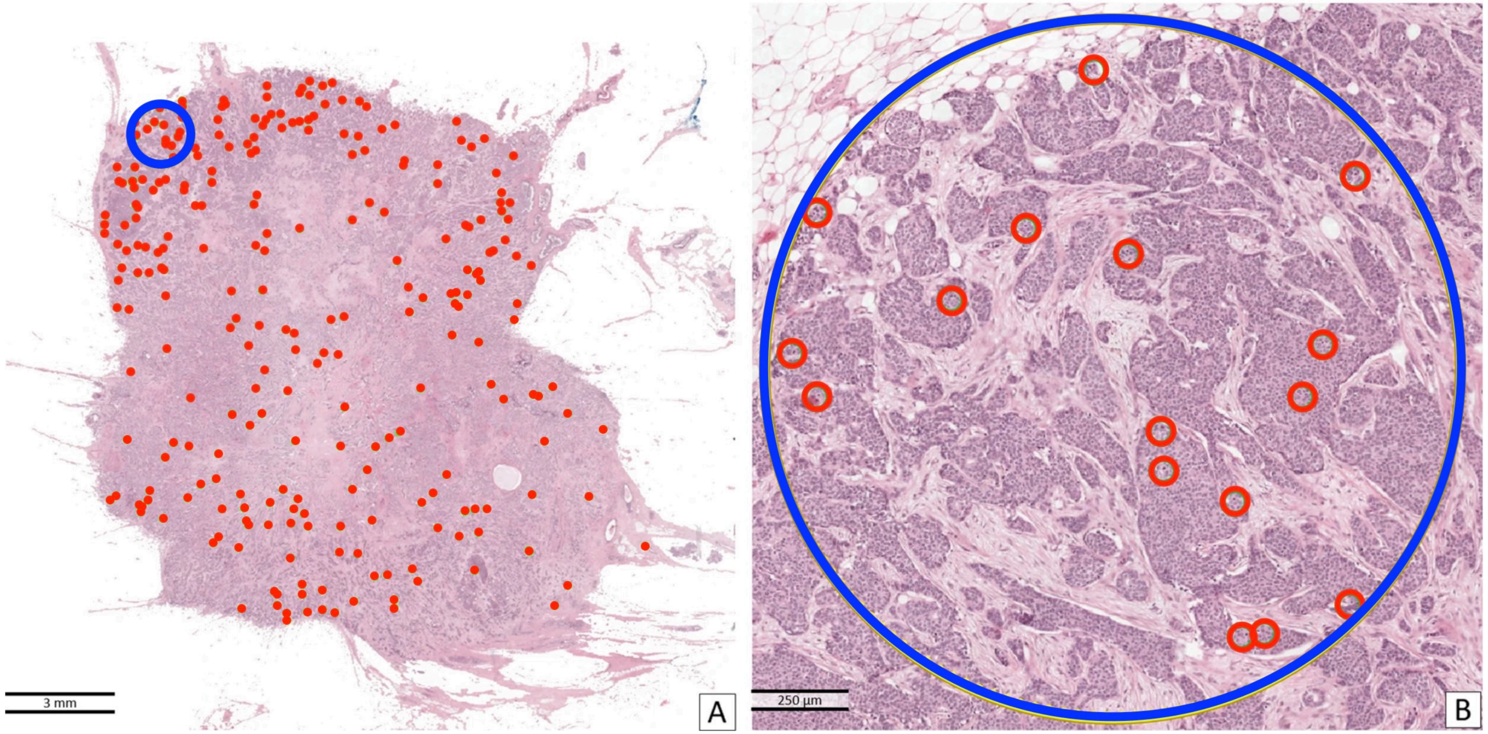
Motivation

- Dense areas of **mitotic cells** are indicative of the presence of breast cancer



Motivation

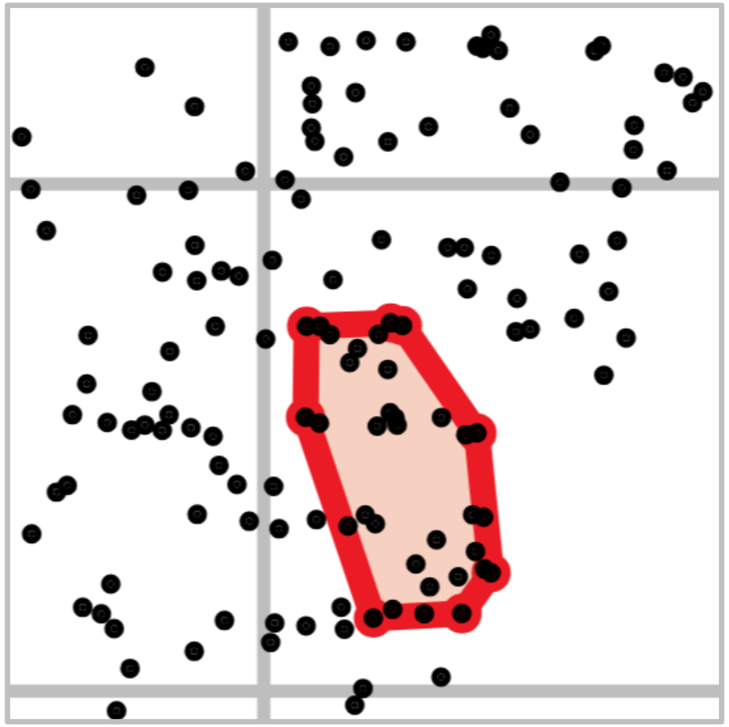
- Pathologists used to evaluate these images by hand
- They mark the mitotic cells as small dots and then locate the **hotspot region**



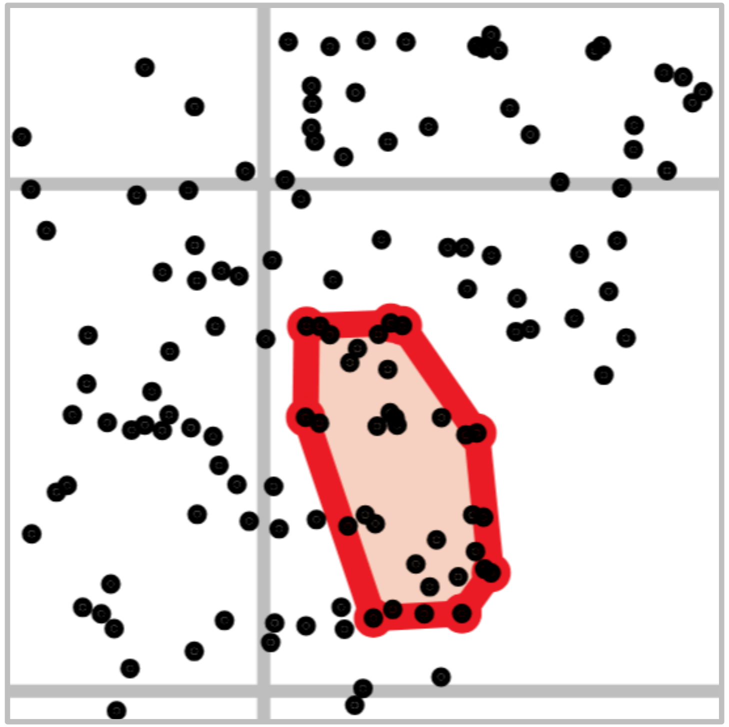
Motivation

- The **hotspot region** satisfies three requirements:

  1. It is convex
  2. It contains the maximum number of points
  3. It has bounded area and diameter (elongation)



Motivation



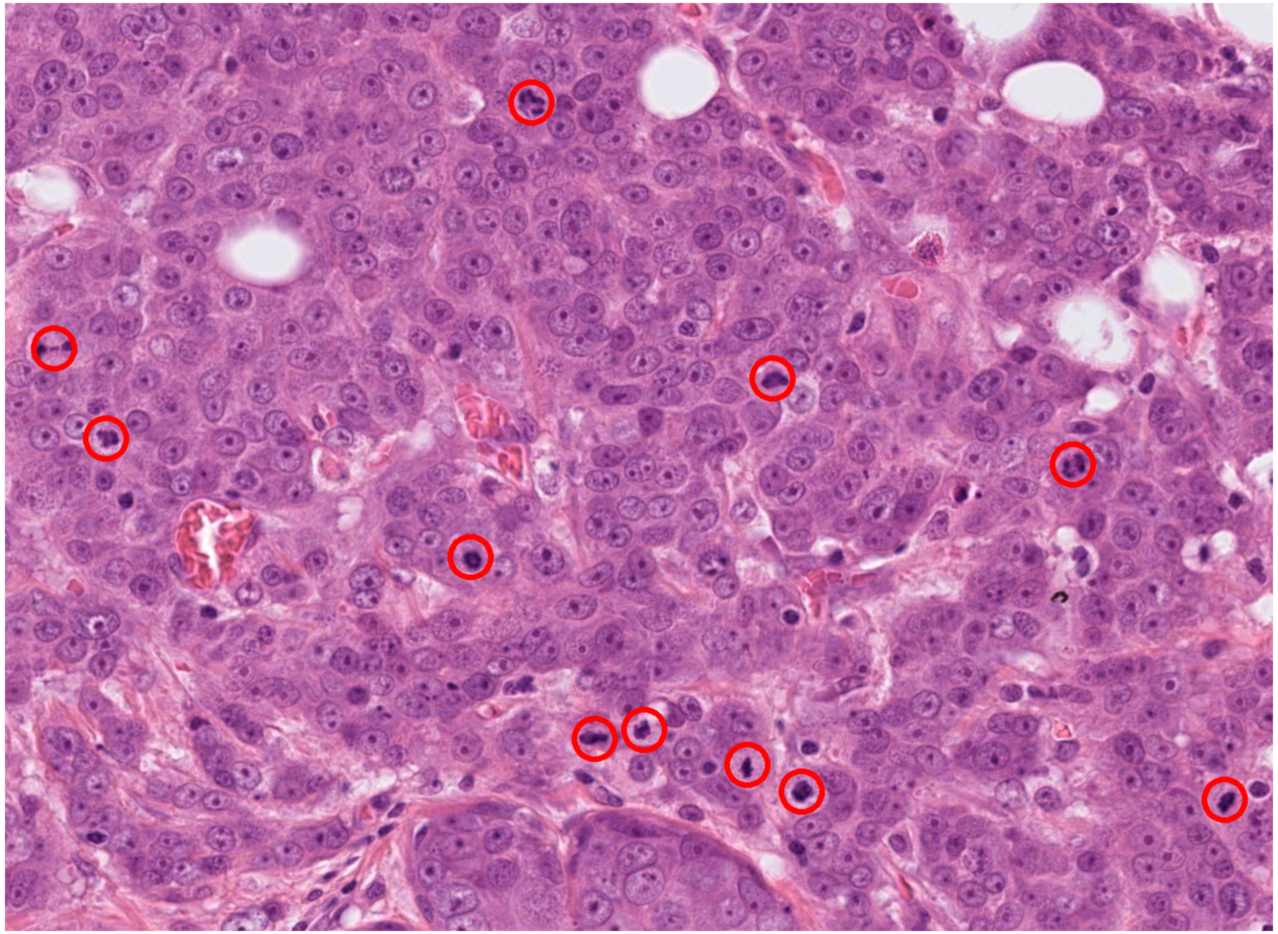
- The **hotspot region** satisfies three requirements:

  1. It is convex
  2. It contains the maximum number of points
  3. It has bounded area and diameter (elongation)

- The number of cells in the hotspot region is called **Mitotic Count** and is used to establish the risk and treatment regime for patients

Motivation

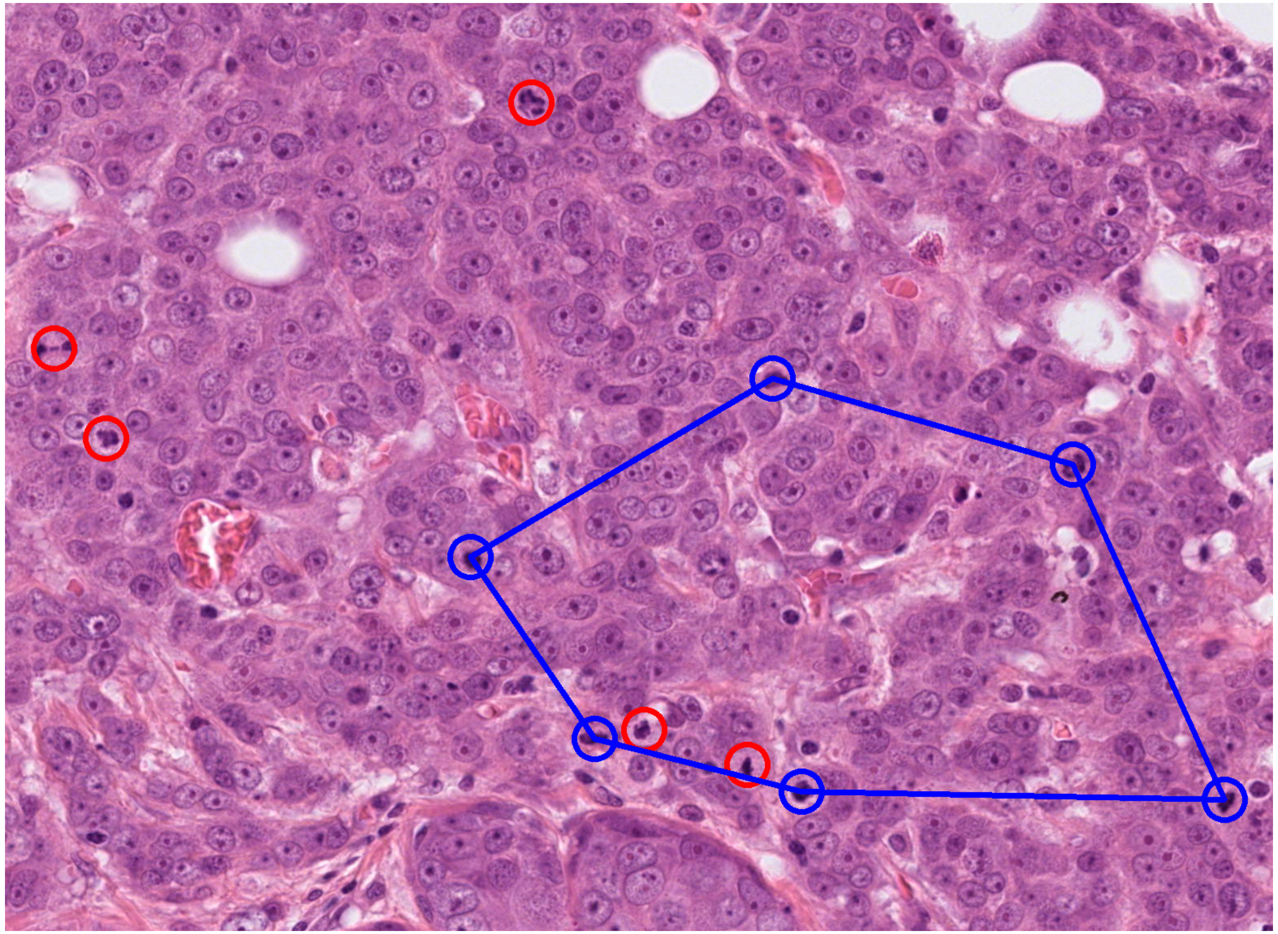
- AI models can already identify mitotic cells with high accuracy



Utrecht University

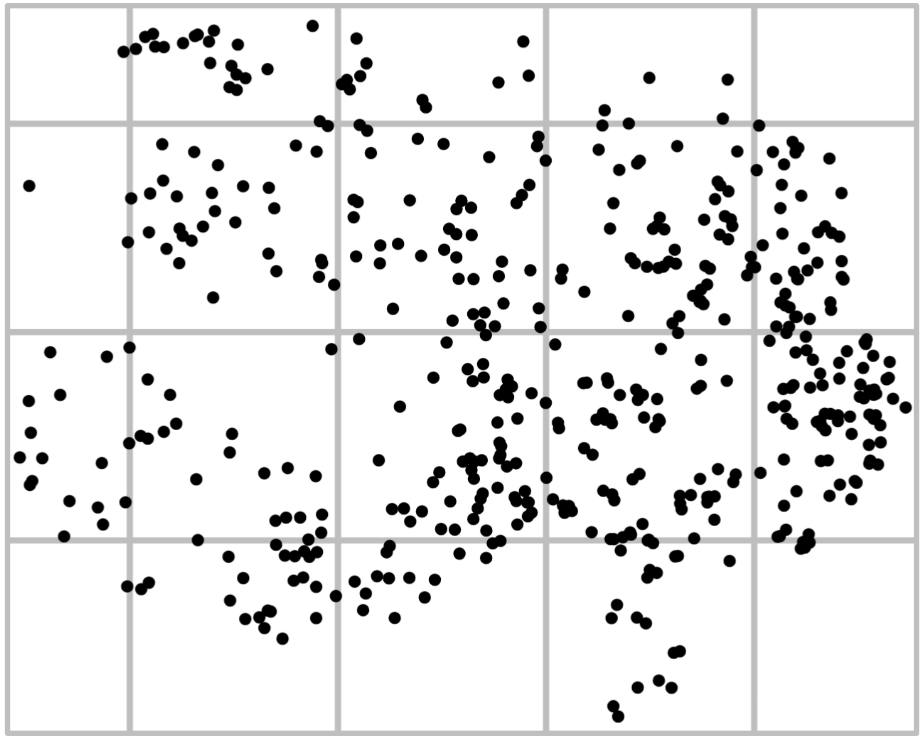
Motivation

- AI models can already identify mitotic cells with high accuracy
- Locating the hotspot region remains a difficult problem



Problem definition

- Given a set of n points, a maximum area A_{max} and diameter D_{max}

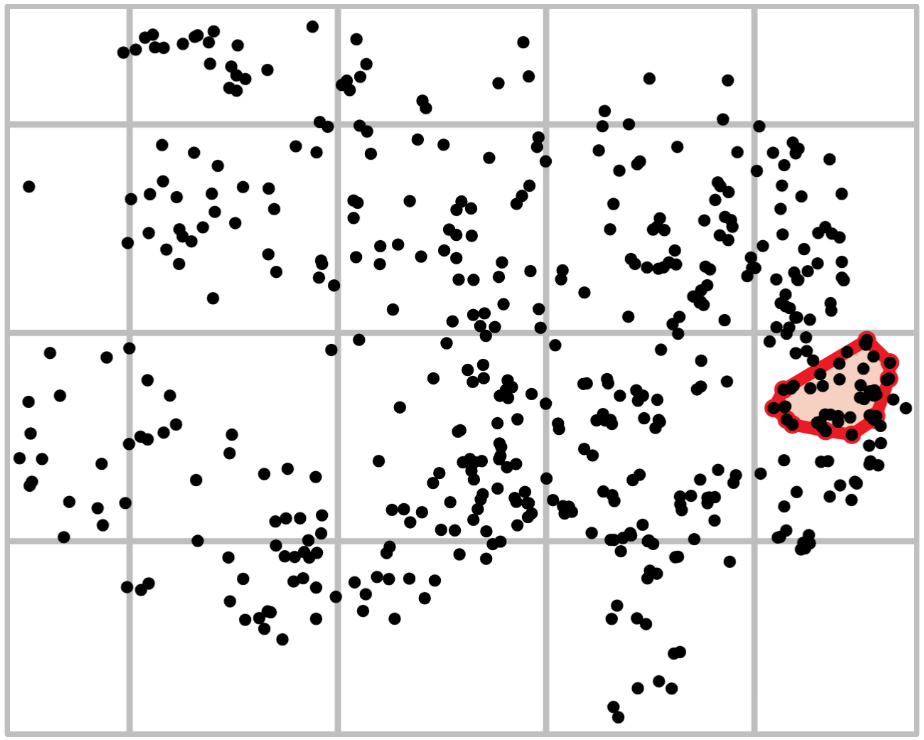


$A_{\text{max}} = 4\text{mm}^2$
$D_{\text{max}} = 4\text{mm}$

Utrecht University

Problem definition

- Given a set of n points, a maximum area $A_{\max}$ and diameter $D_{\max}$
- Find the hotspot region with area $\leq A_{\max}$ and diameter $\leq D_{\max}$



$A_{\max} = 4\text{mm}^2$

$D_{\max} = 4\text{mm}$

Utrecht University

Previous work

	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5}\log k + n\log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Previous work

	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5} \log k + n \log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Previous work

	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5}\log k + n\log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Previous work

	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5}\log k + n\log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Previous work

	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5} \log k + n \log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Previous work

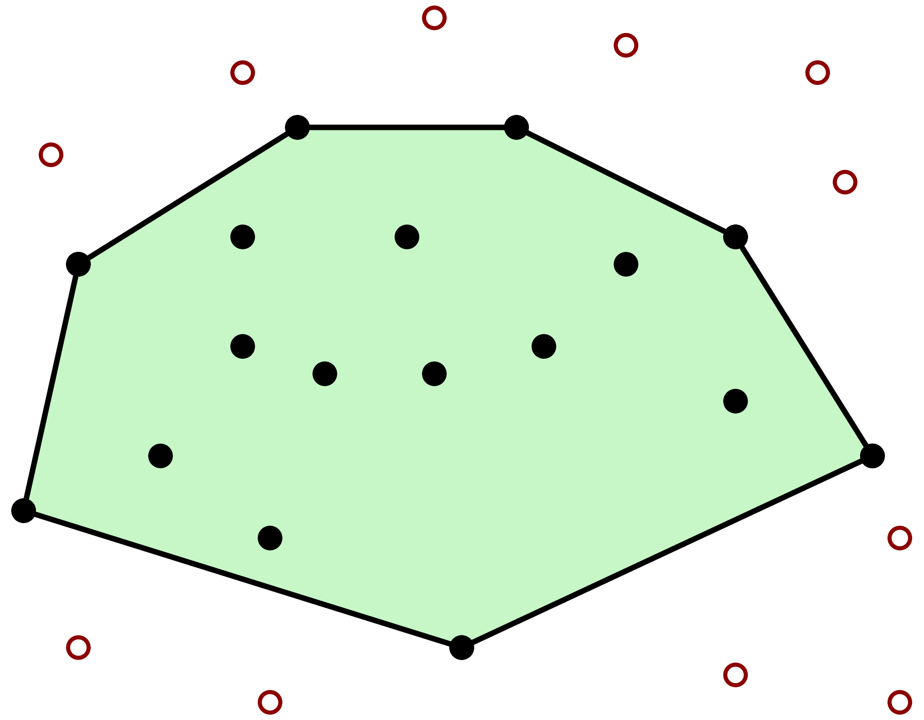
	Eppstein et al. Algorithm A	Aggarwal et al.	Stathonikos et al. Algorithm AS	Algorithm AD (ours)
Bounds	Area	Diameter	Area & Diameter	Area & Diameter
Optimal solution	Yes	Yes	No	Yes
Runtime	$O(kn^3)$	$O(nk^{2.5}\log k + n\log n)$	$O(n2^n)$ or $O(n^3)$	$O(kn^6)$

Algorithm Area-only (A)

By Eppstein et al.

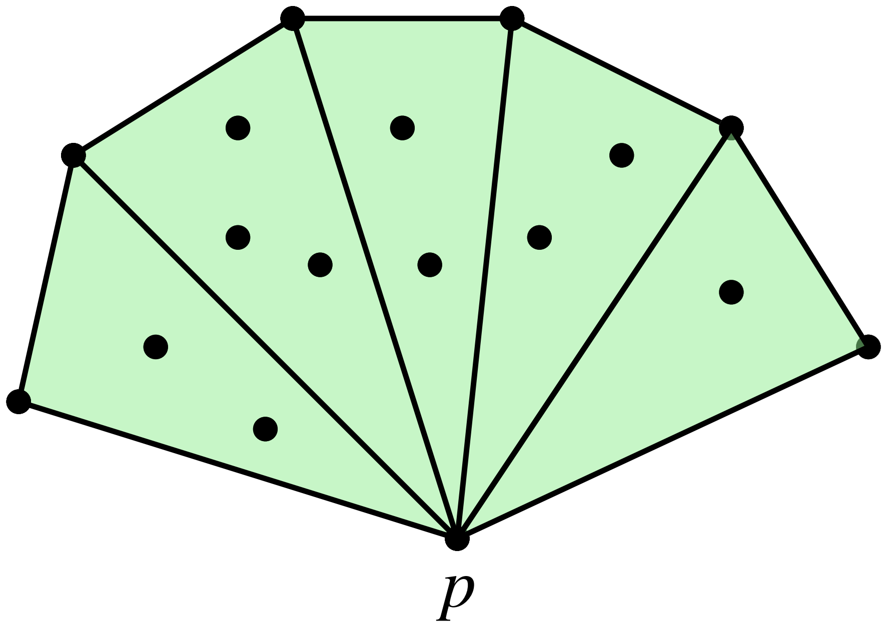
Algorithm A

- Suppose this is the optimal region with k points



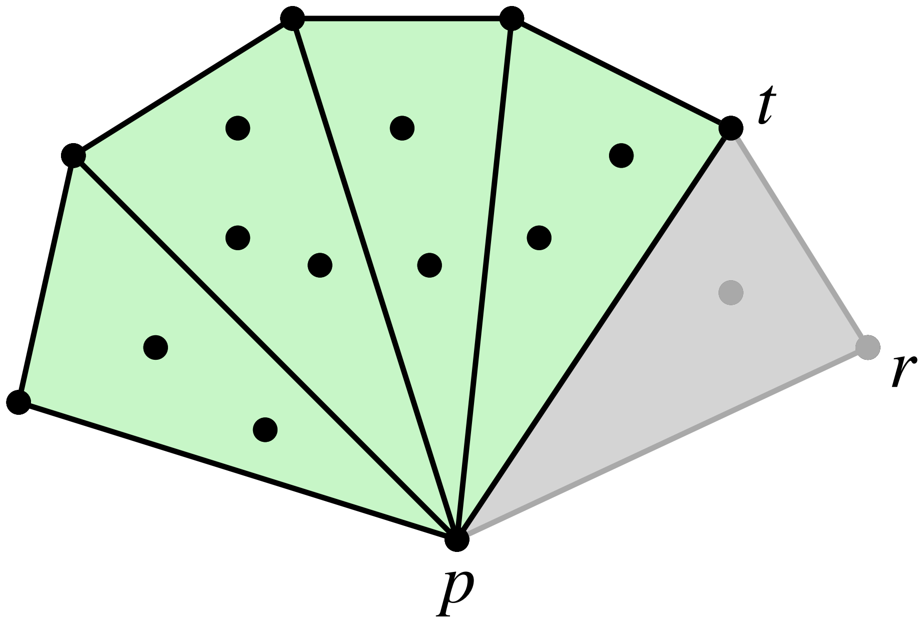
Algorithm A

- Clockwise fan of triangles centred at p



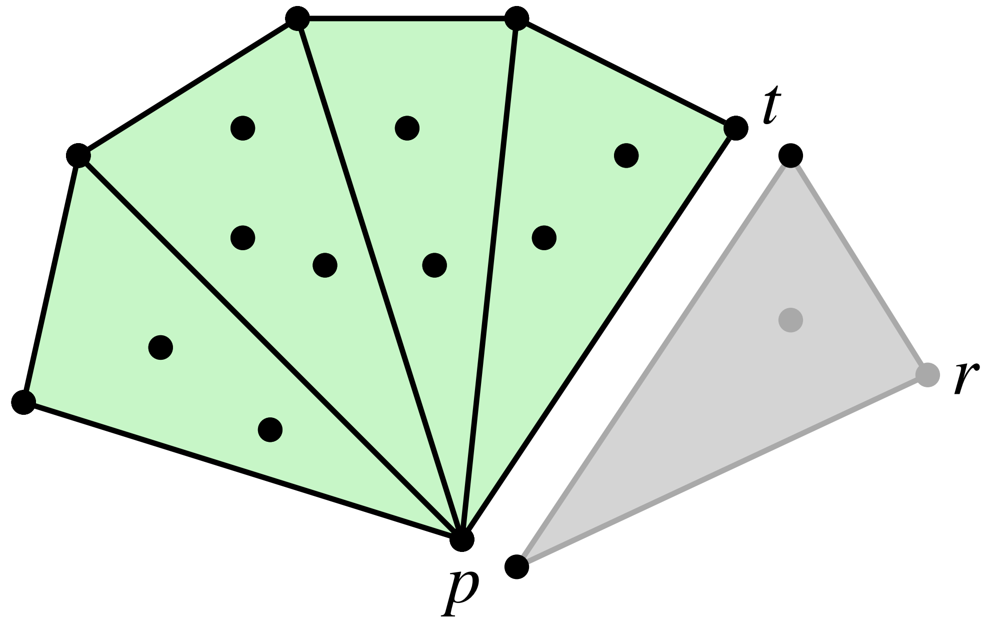
Algorithm A

- Clockwise fan of triangles centred at p
- Remove the most clockwise triangle



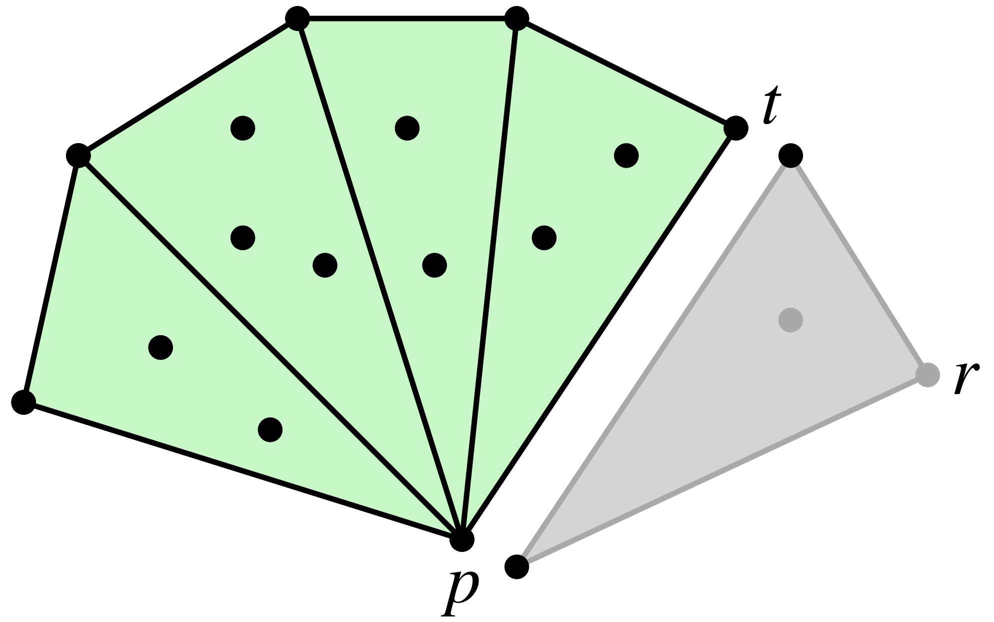
Algorithm A

- Clockwise fan of triangles centred at p
- Remove the most clockwise triangle



Algorithm A

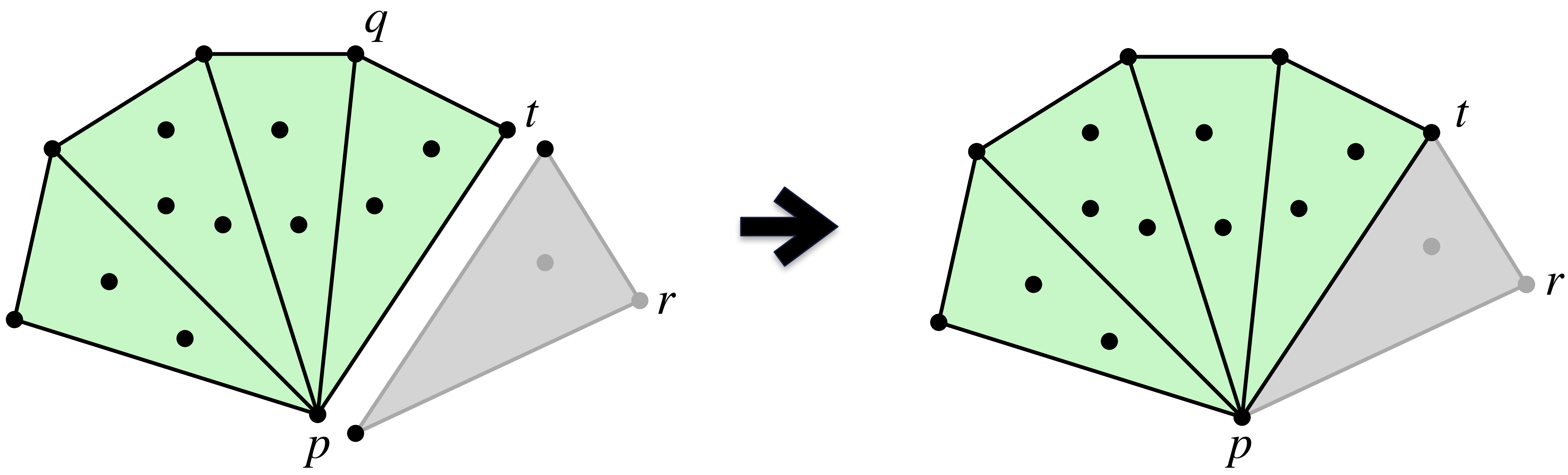
- Clockwise fan of triangles centred at p
- Remove the most clockwise triangle
- Region still optimal with $k - K(p, r, t)$ points



Utrecht University

Algorithm A

$$T[r,t,k] = A(p,r,t) + \min_{q} T[r,q,k-K(p,r,t)]$$

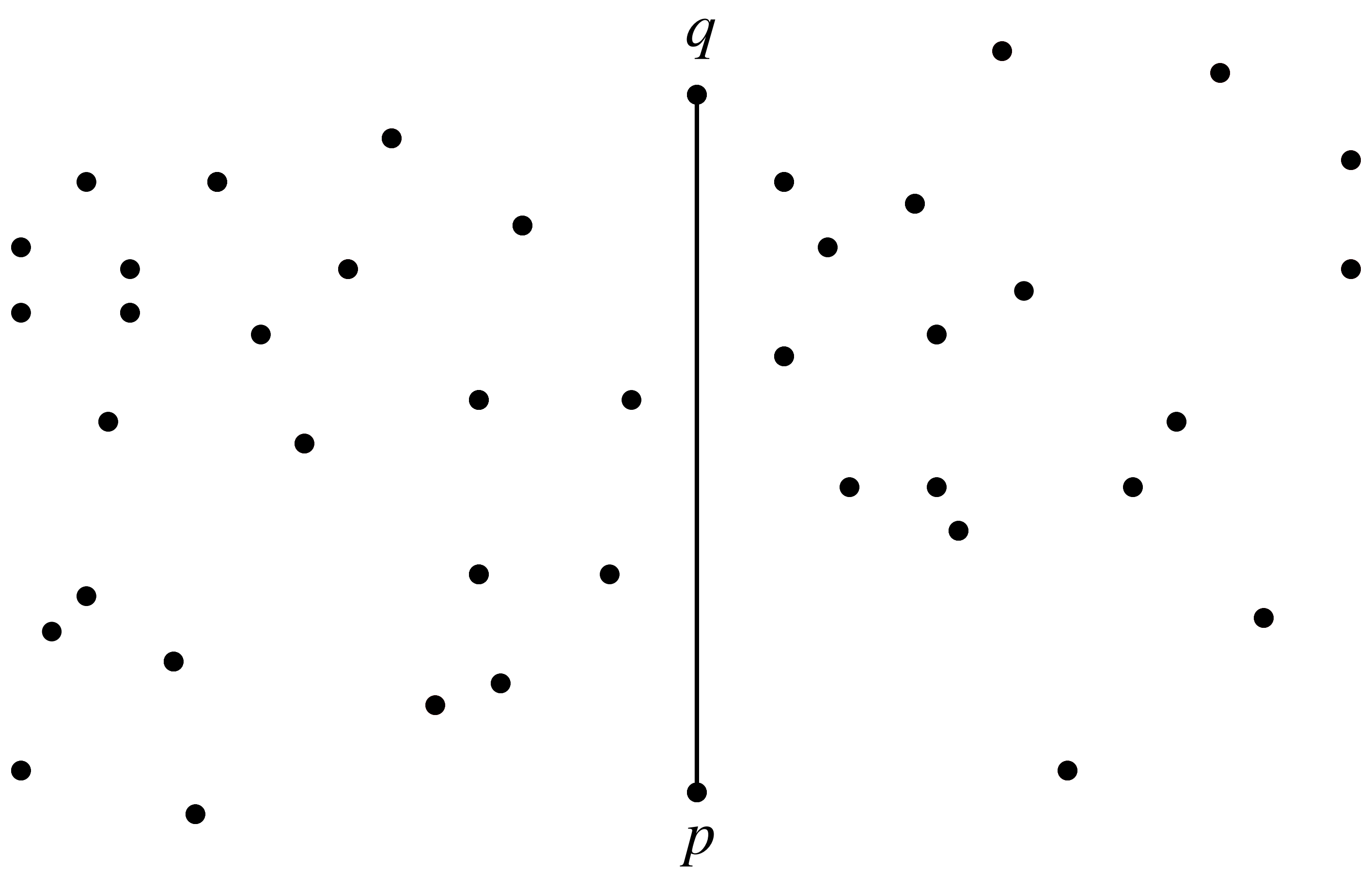


Algorithm Area–Diameter (AD)

Our contribution

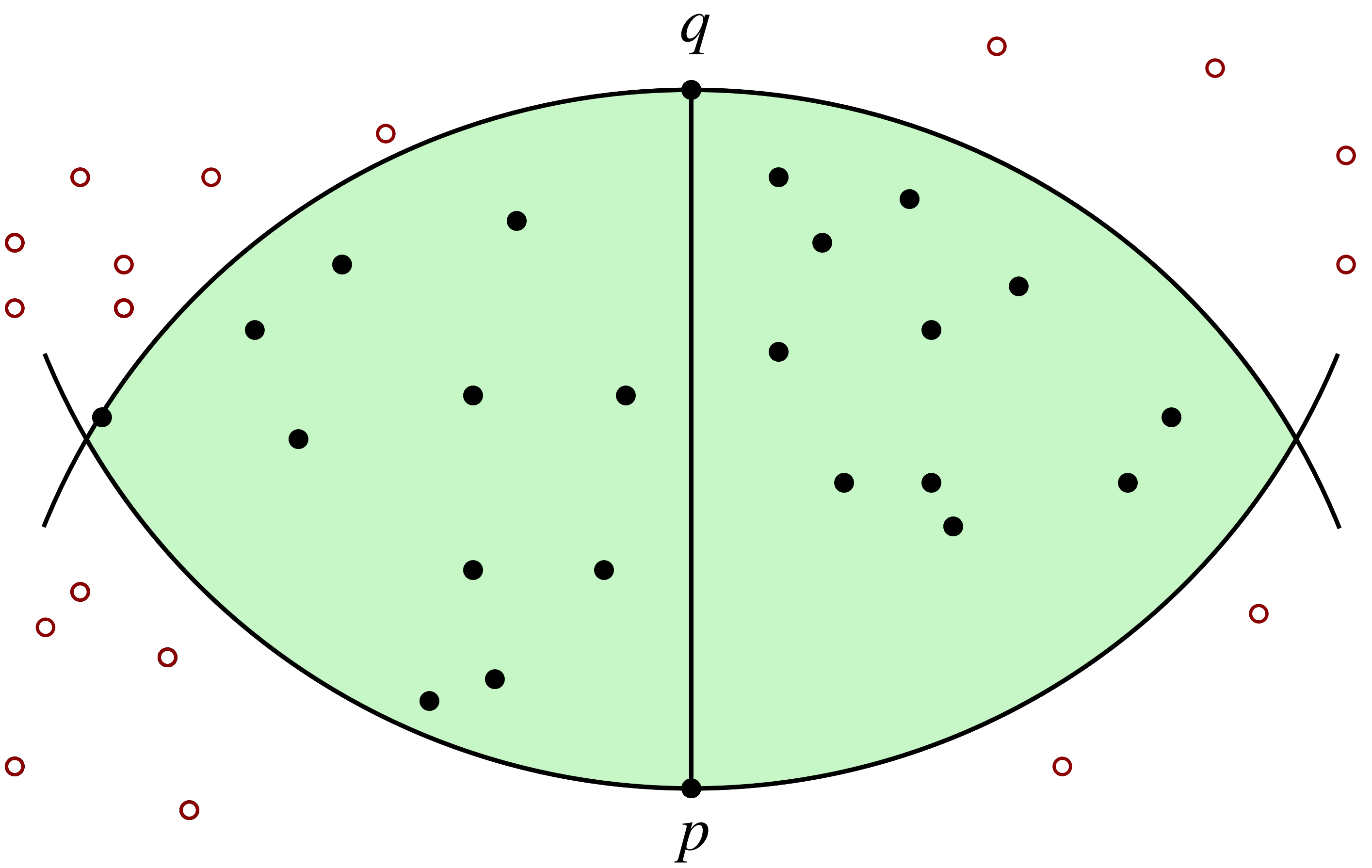
Algorithm AD

- Assume the points p, q define the diametral pair of the optimal region



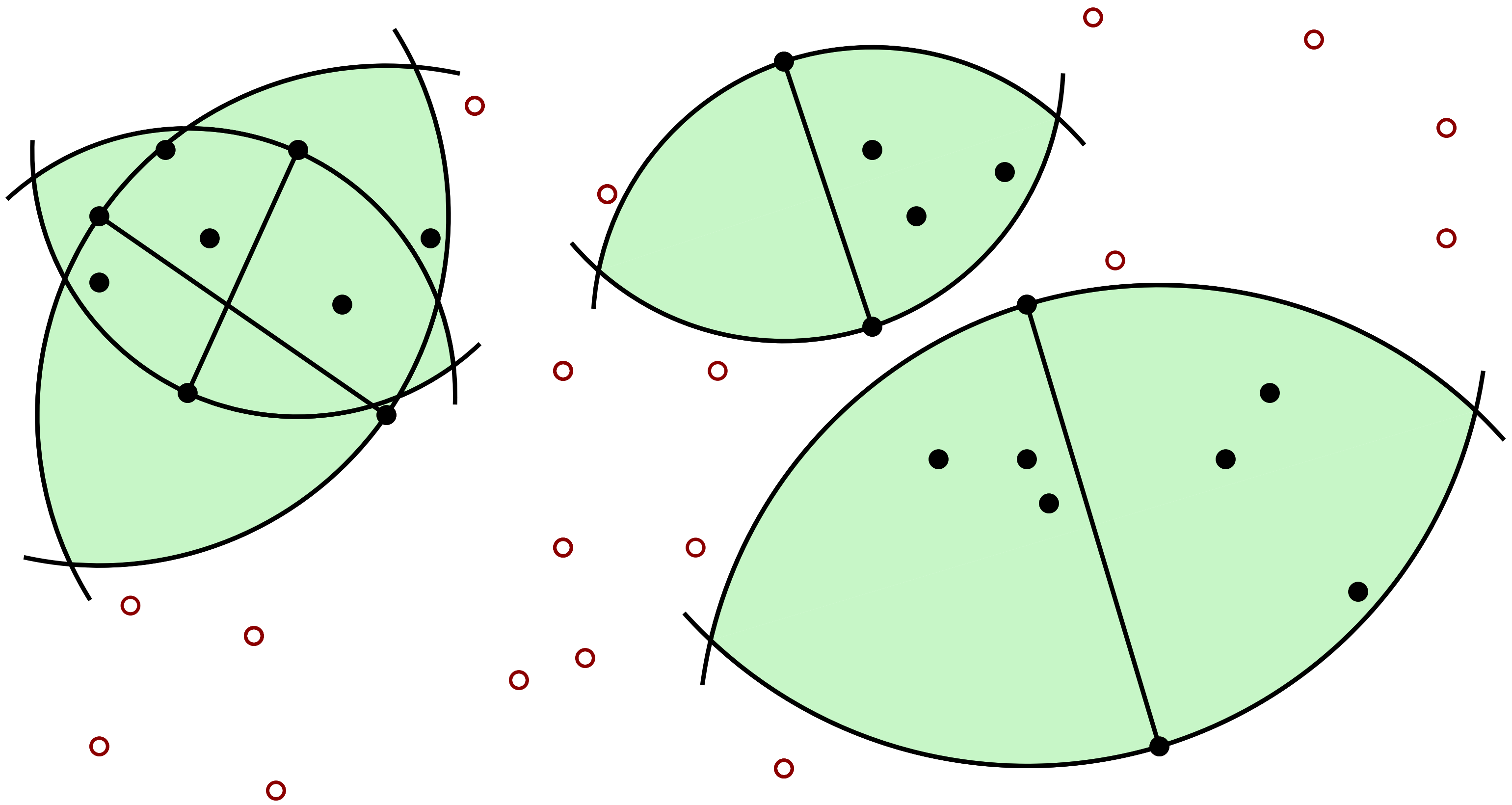
Algorithm AD

- The optimal region must be defined by points no further than $|pq|$ from p and q



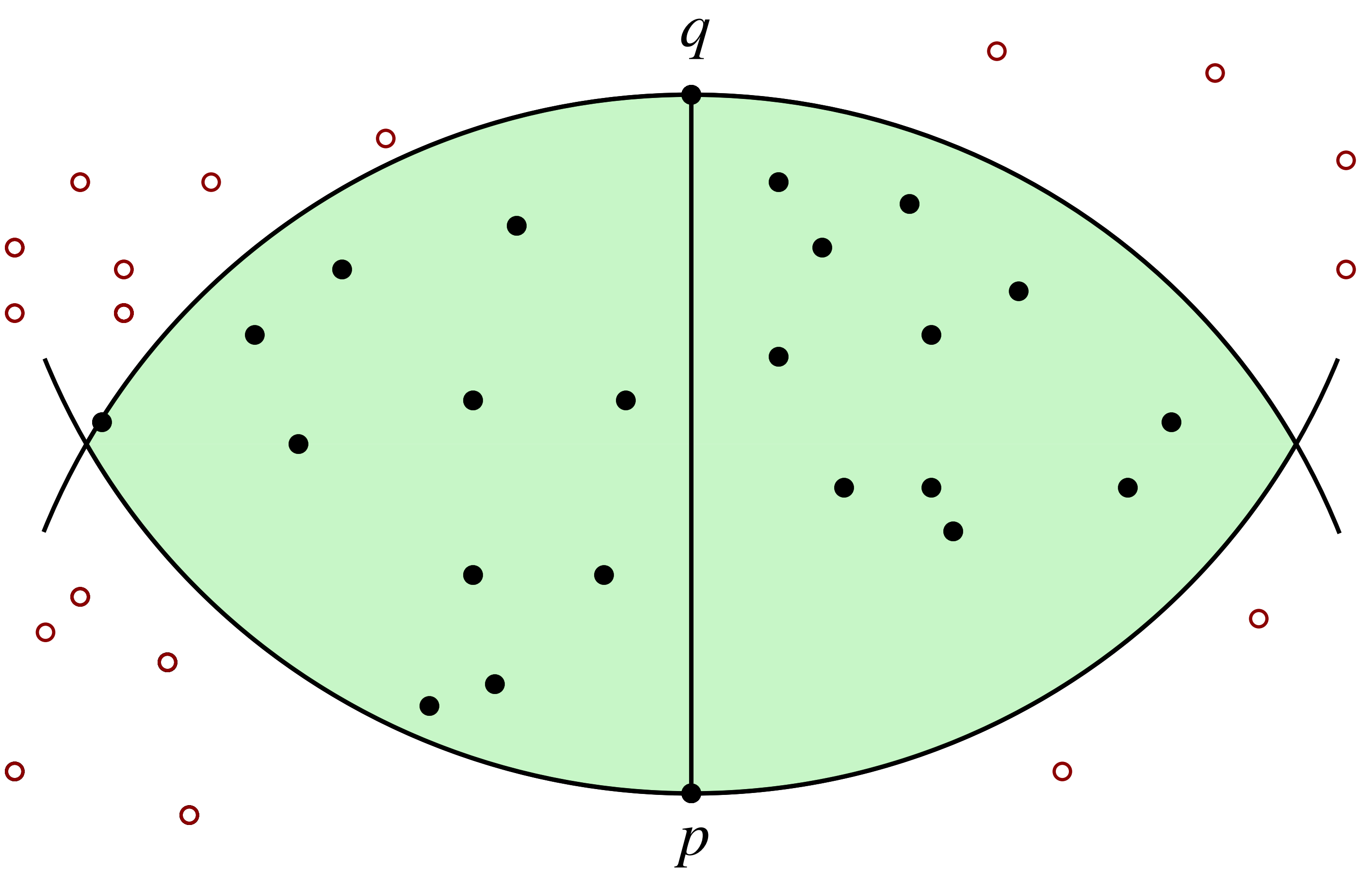
Algorithm AD

- The algorithm processes up to $O(n^2)$ diametral pairs



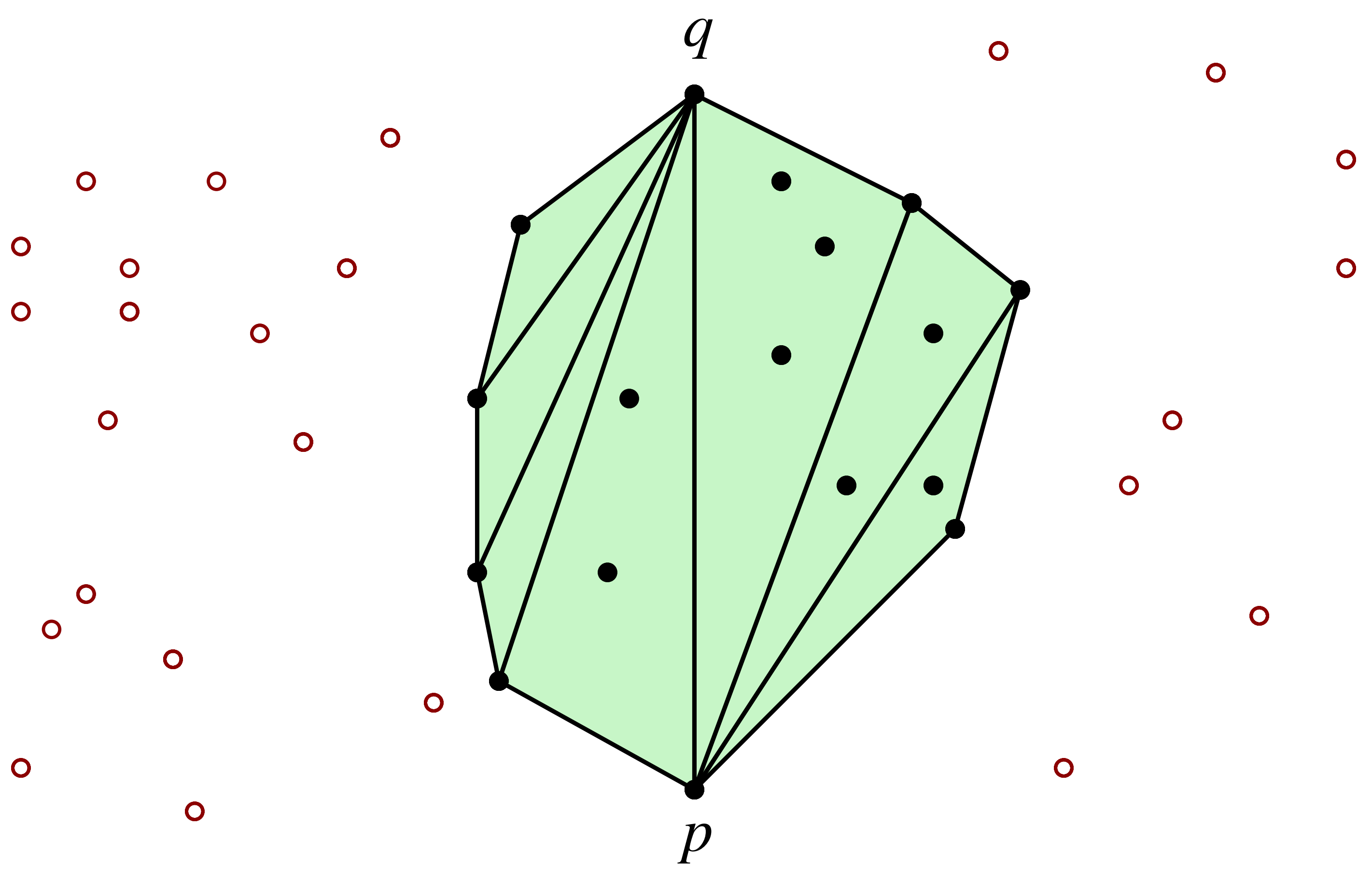
Algorithm AD

- The optimal region is constructed by adding triangles in $O(kn^4)$ time



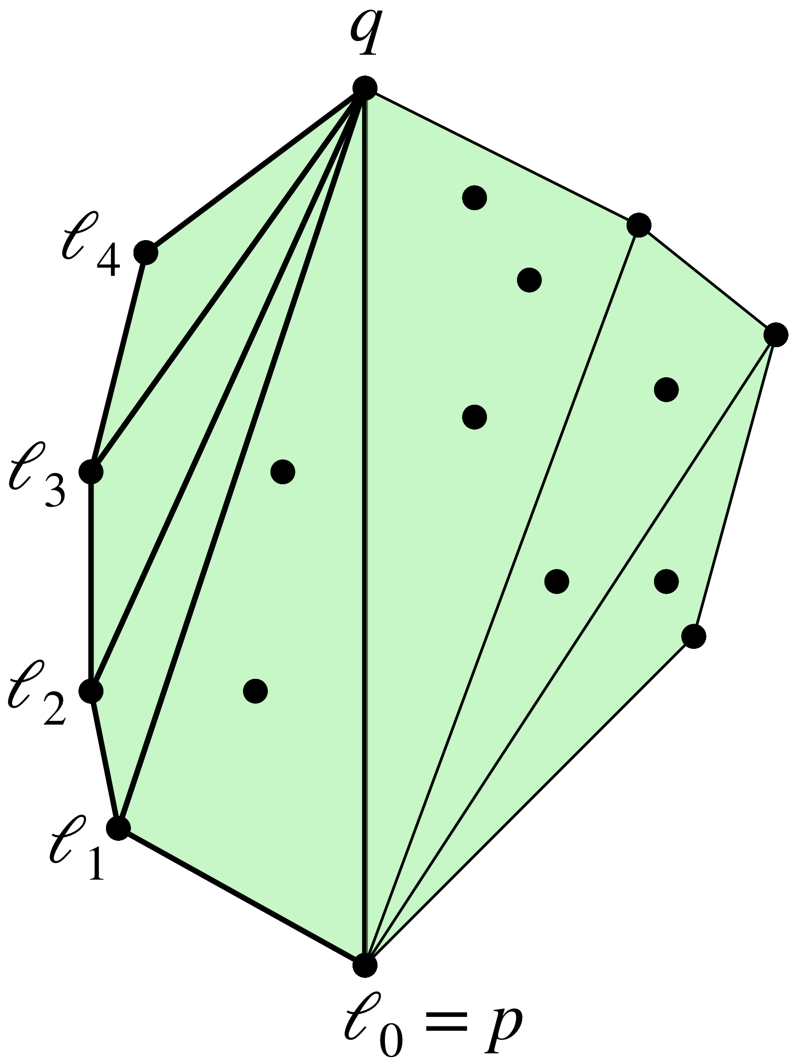
Algorithm AD

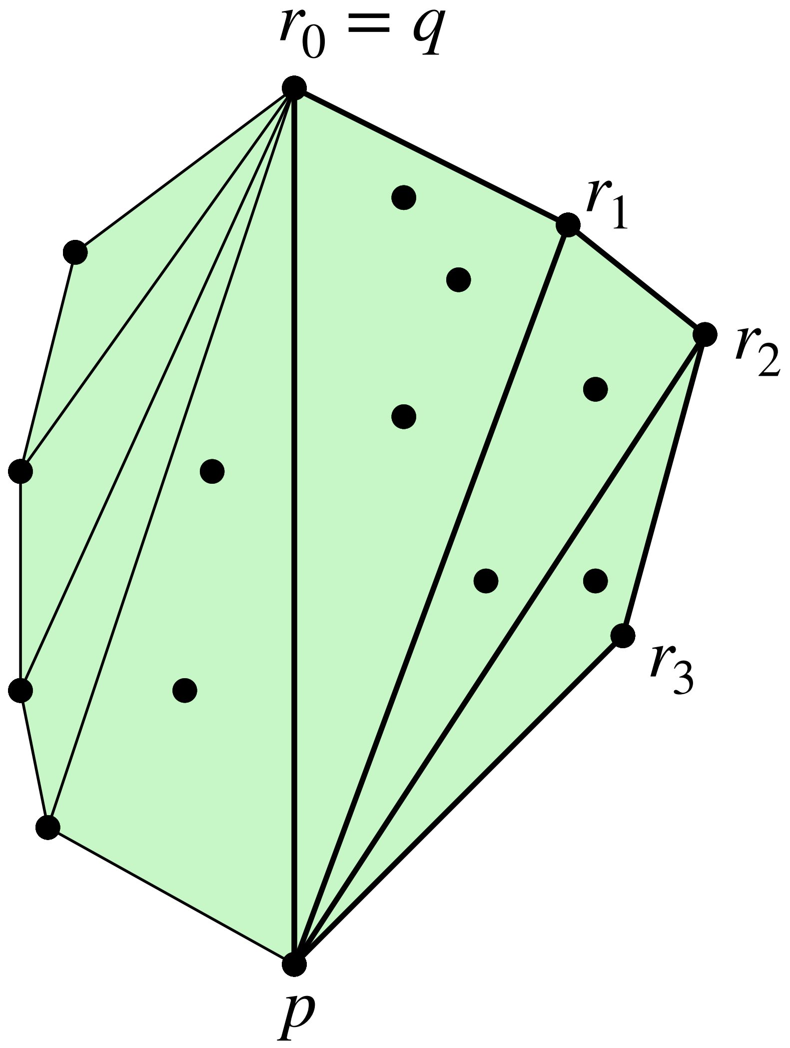
- The optimal region is constructed by adding triangles in $O(kn^4)$ time



Utrecht University

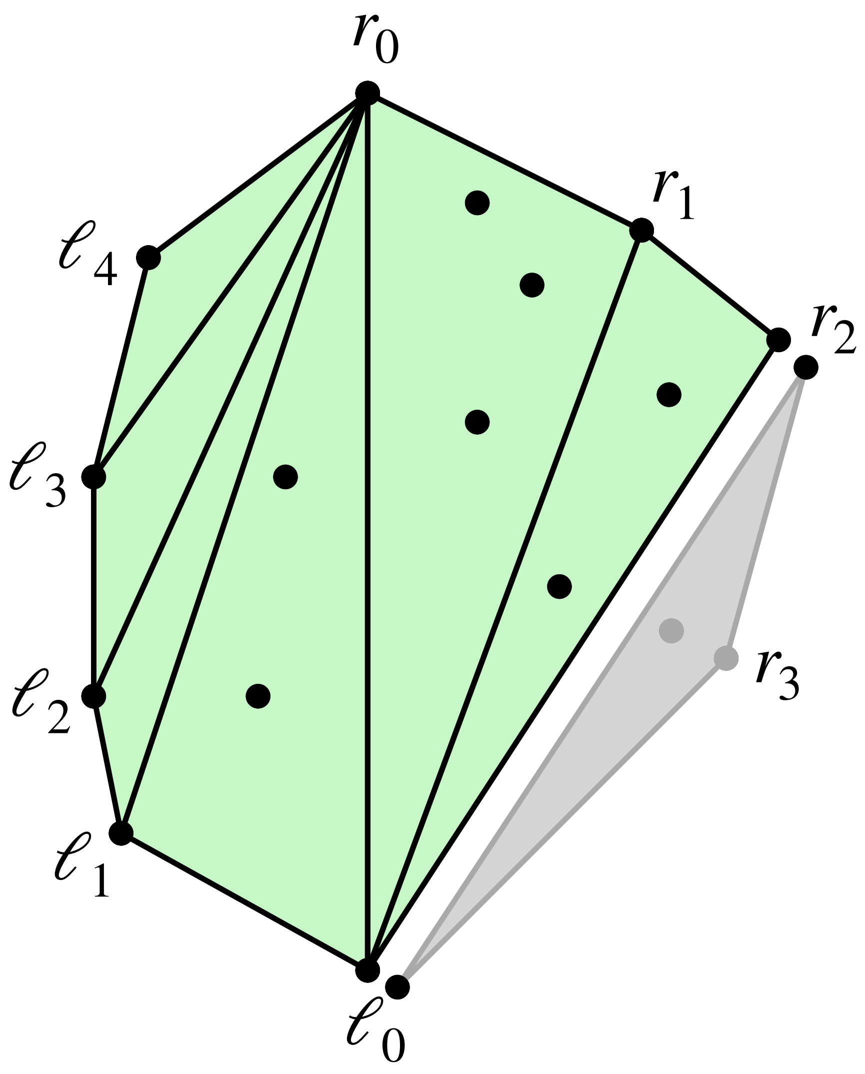
Algorithm AD



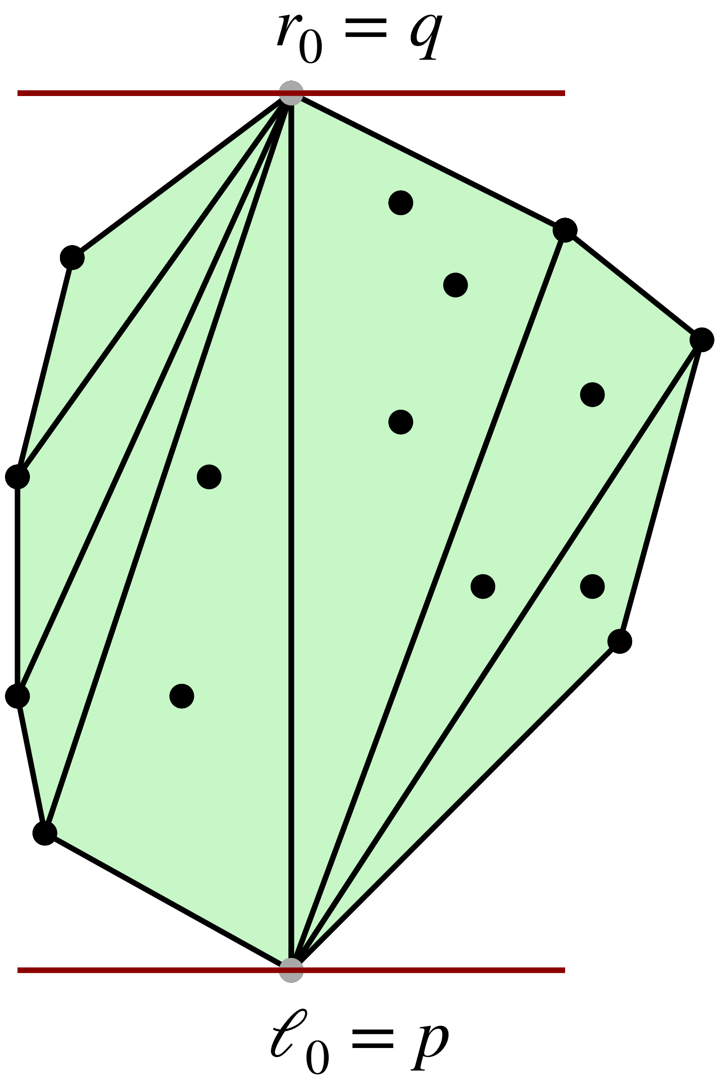


Algorithm AD

- Same optimal substructure as in Algorithm A
- Remove the latest added vertex r_3
- Optimal region with $k - K(\ell_0, r_3, r_2)$ points

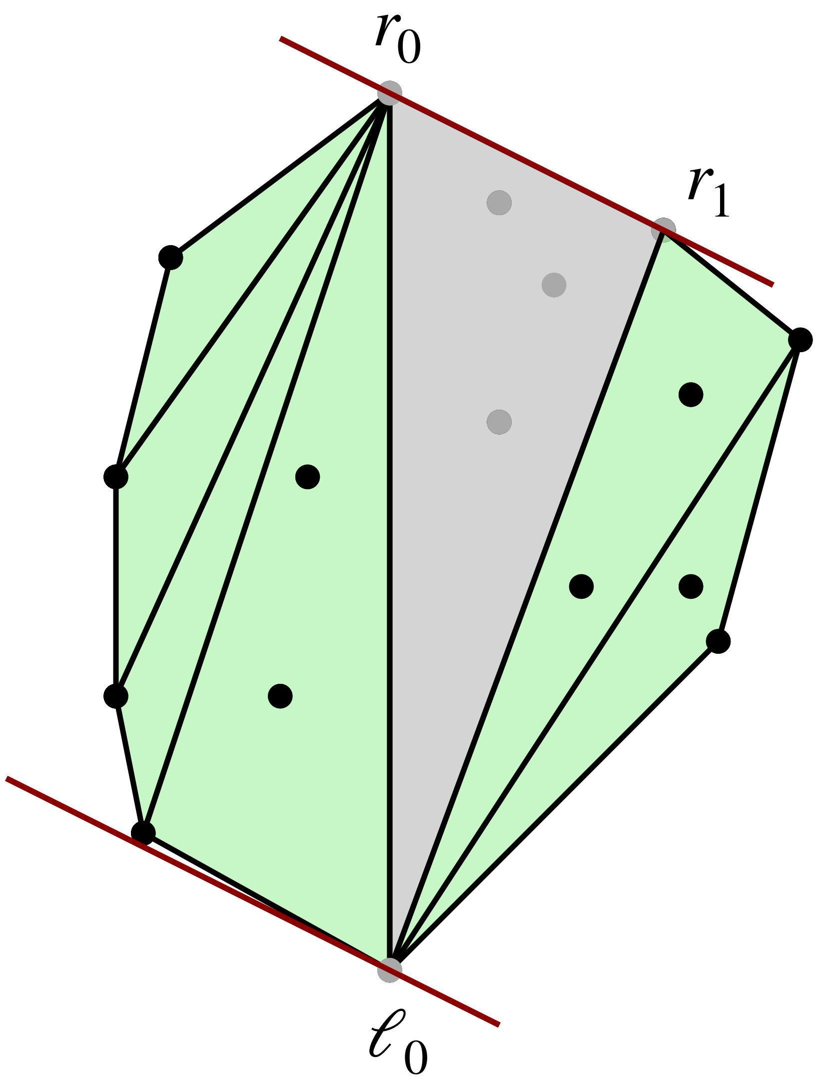


Algorithm AD



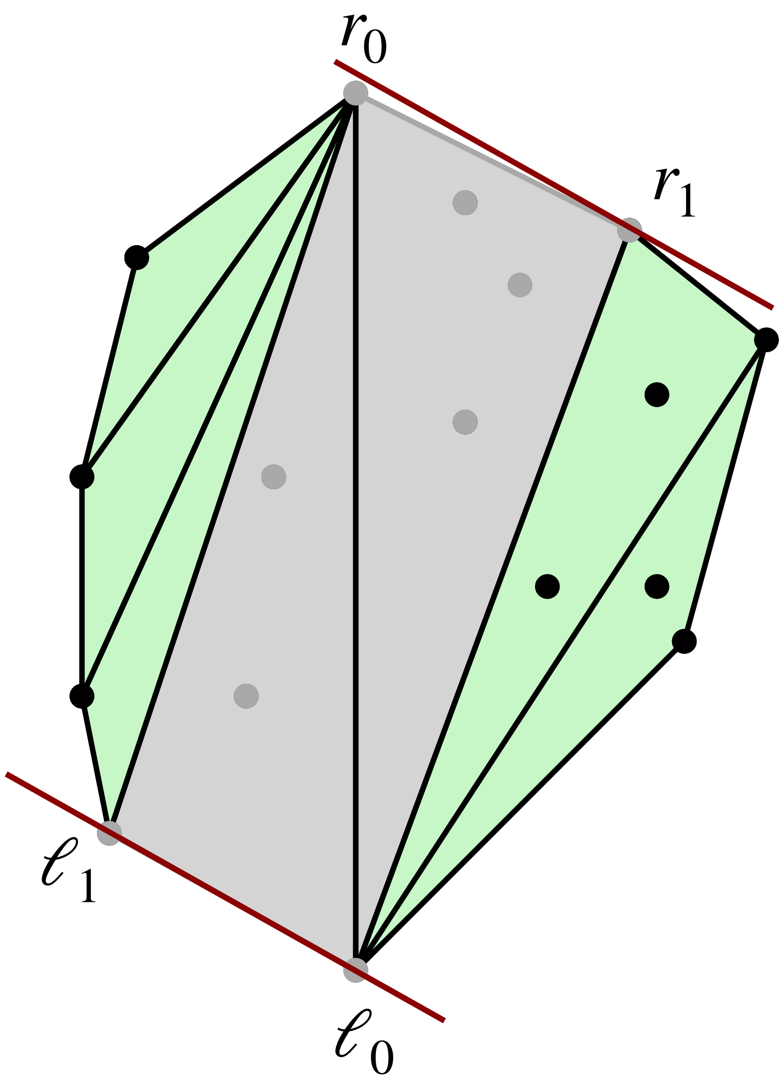
Utrecht University

Algorithm AD



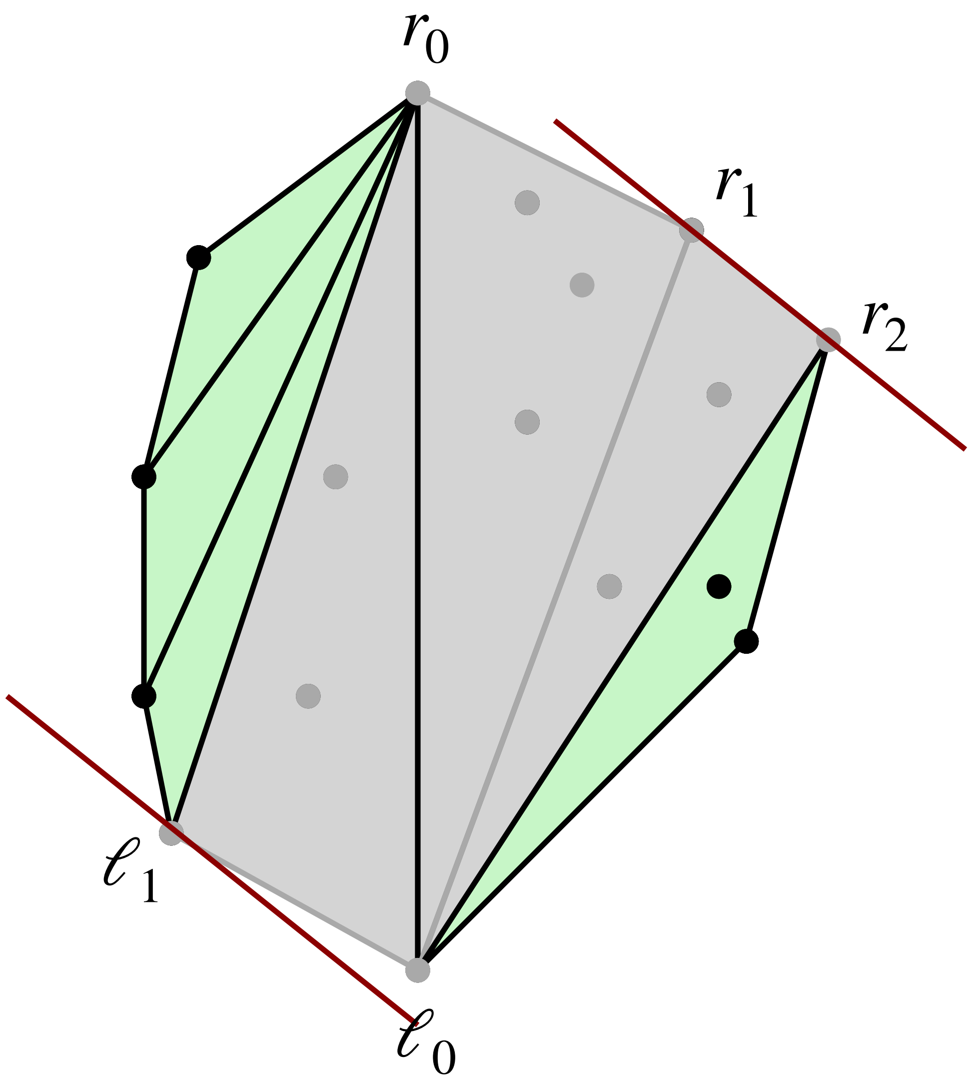
Utrecht University

Algorithm AD

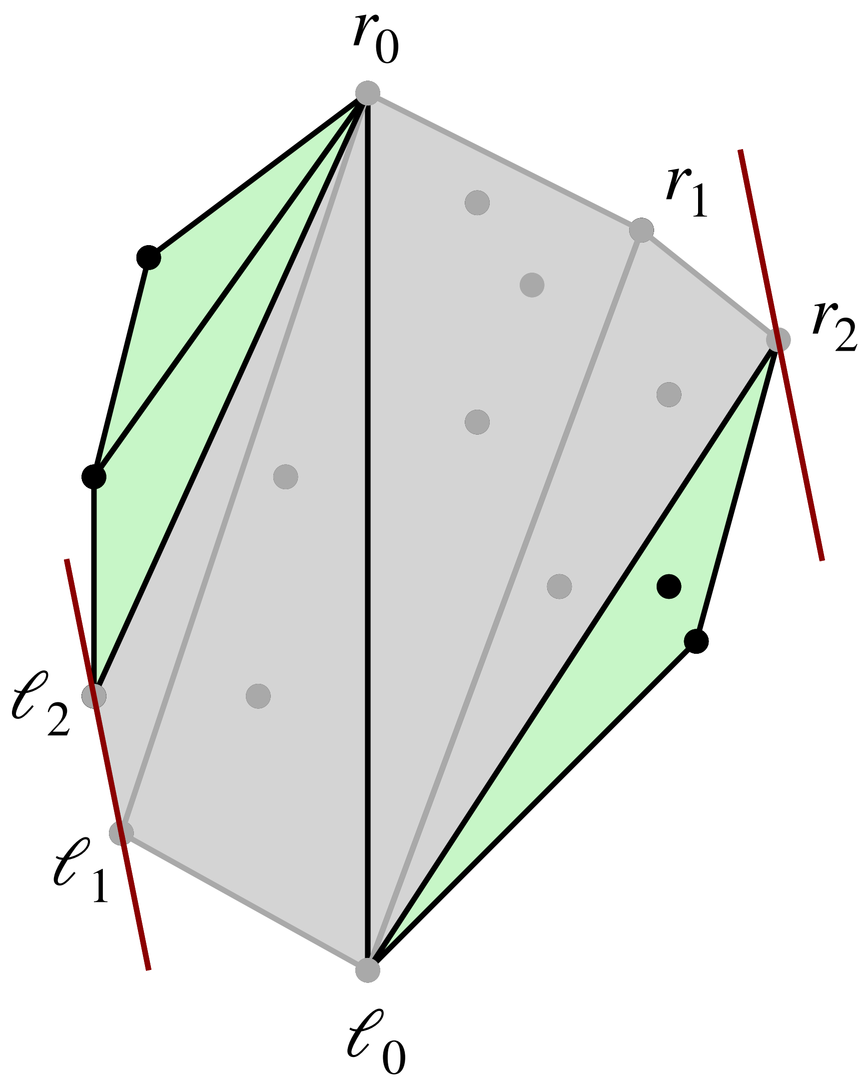


Utrecht University

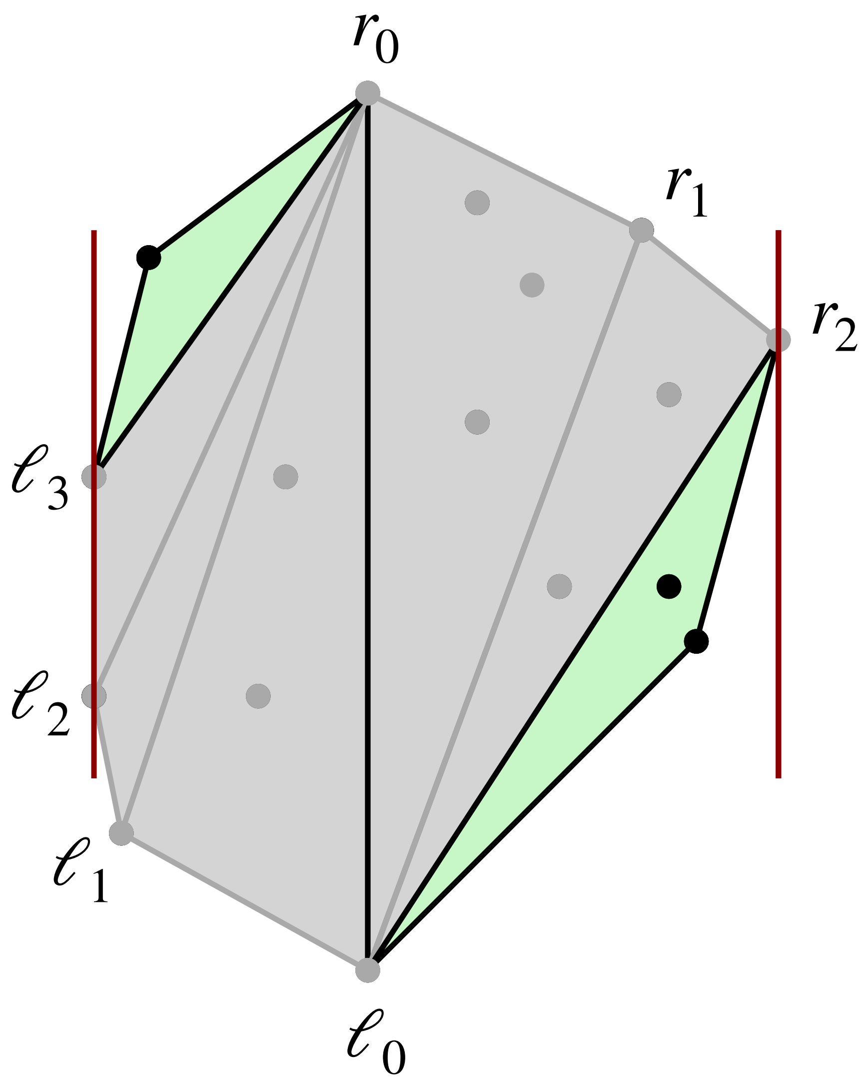
Algorithm AD



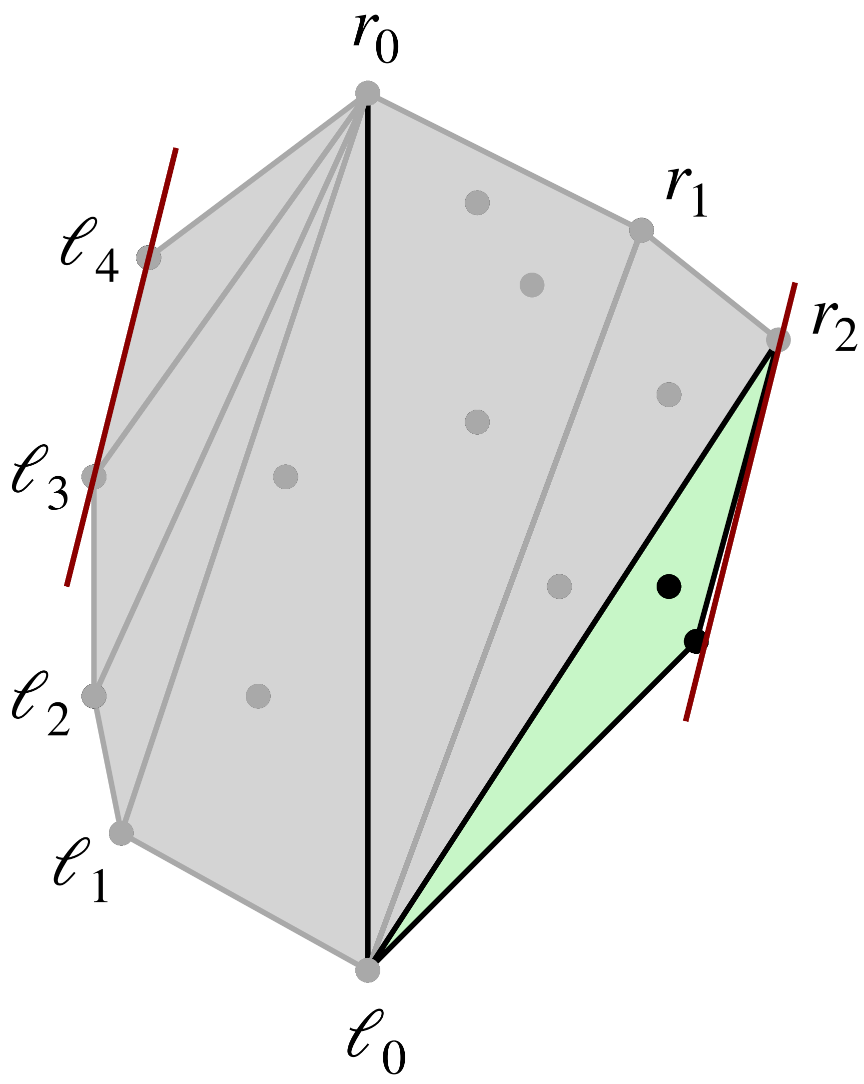
Algorithm AD



Algorithm AD

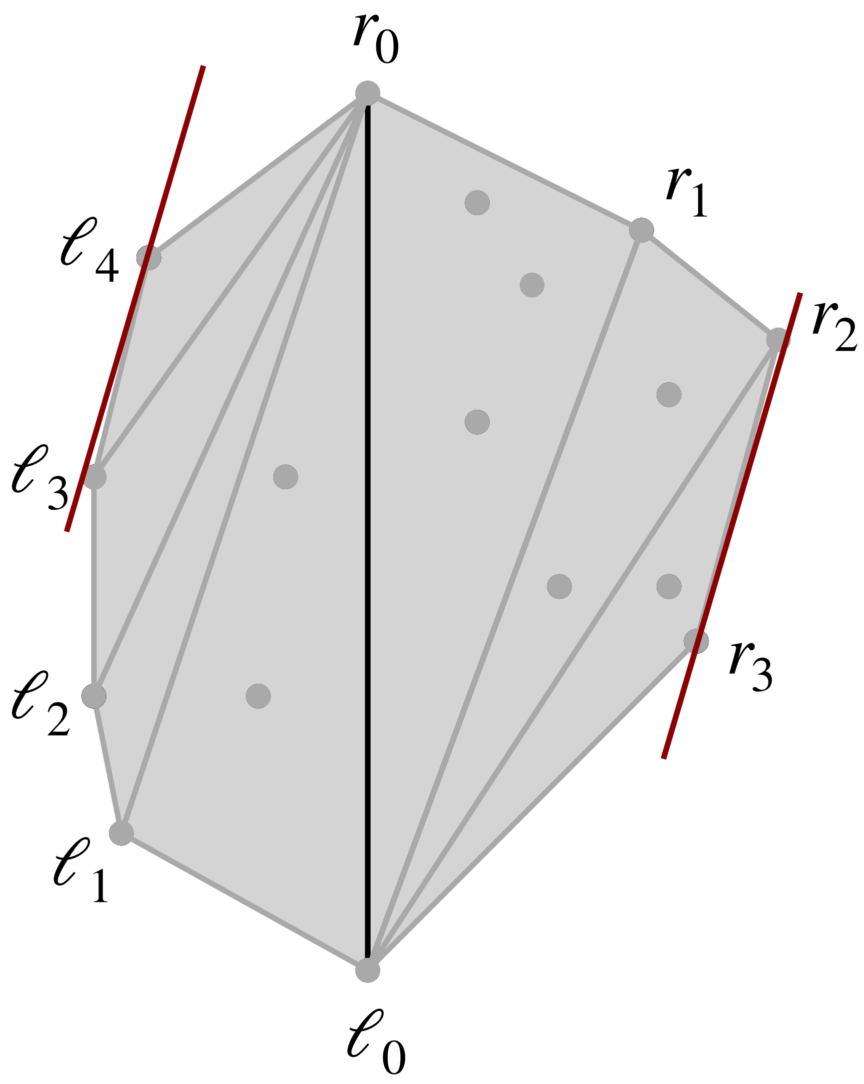


Algorithm AD



Utrecht University

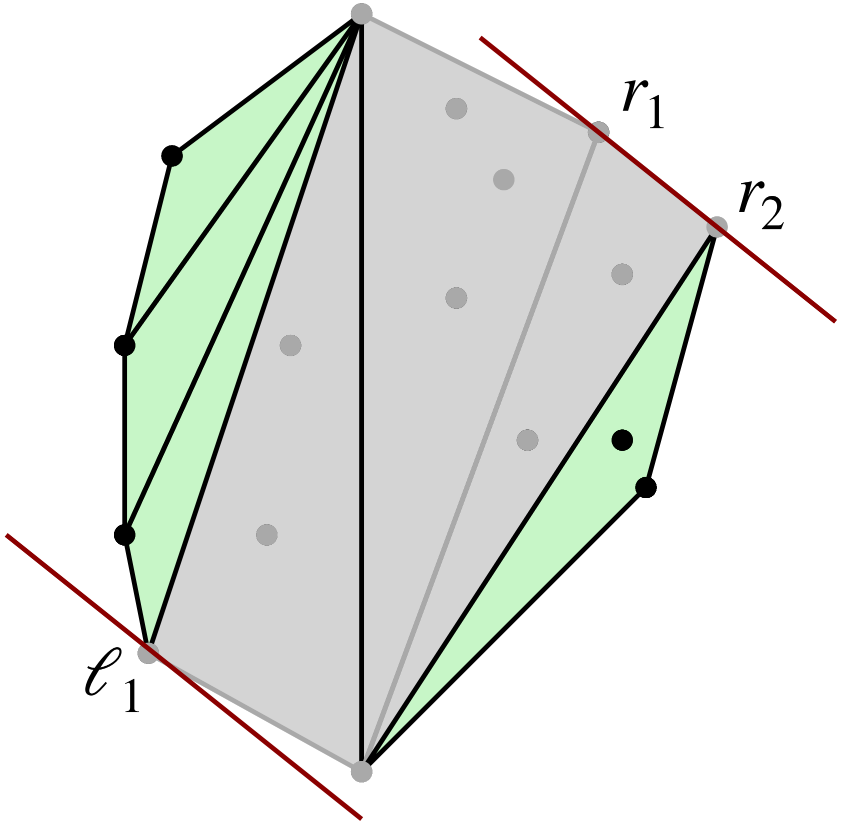
Algorithm AD



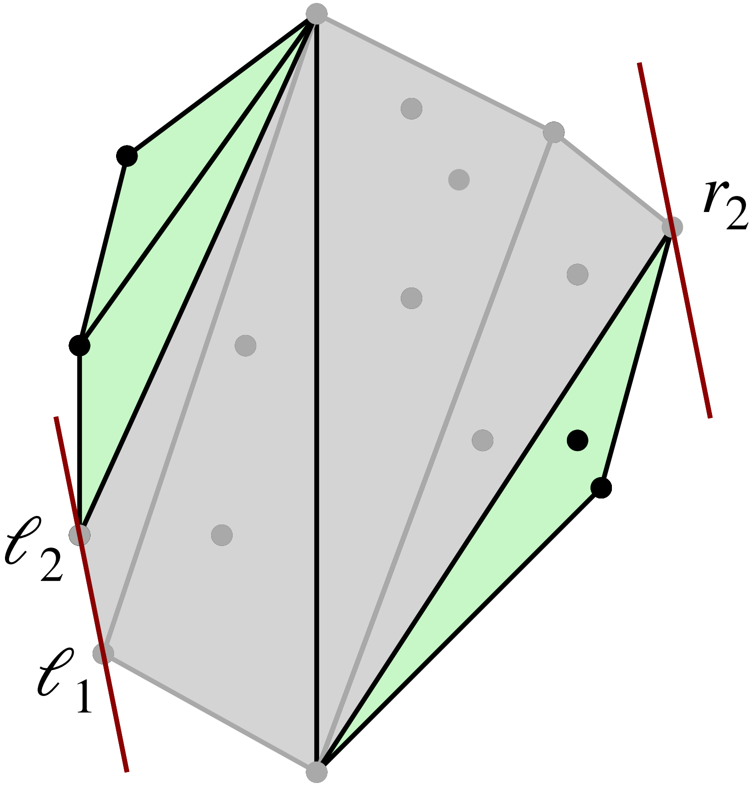
Algorithm AD

- Regions are stored and reused for expansion with new antipodal pairs
- Indexing using two four-dimensional tables

Algorithm AD



$R[r_2, r_1, \ell_1, m = 12]$



$L[\ell_2, \ell_1, r_2, m = 13]$

Algorithm AD

- We prune many unnecessary pairs

    1. Distance-based: exceeding the diameter constraint
    2. Cardinality-based: insufficient number of points in the lune

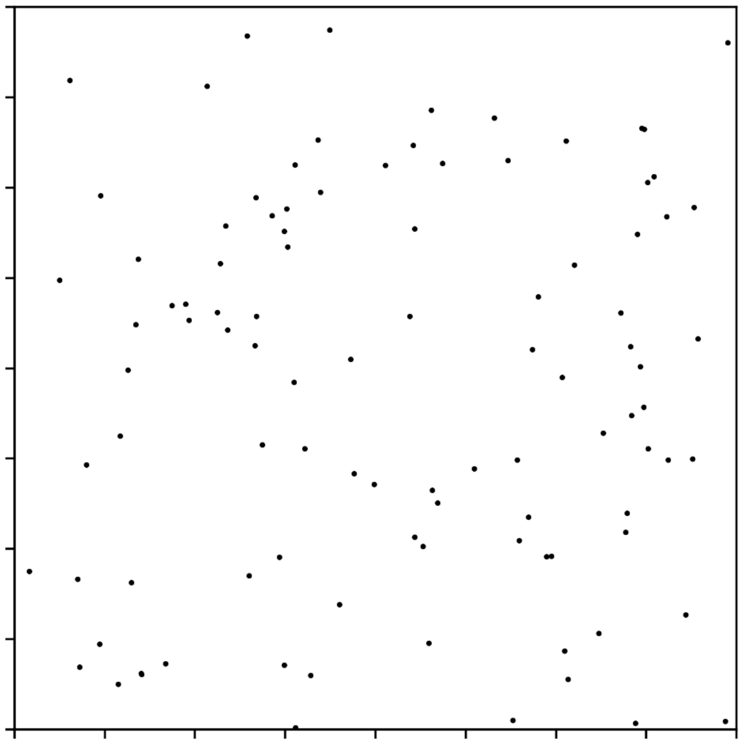
Experimental setup & results

Experimental setup

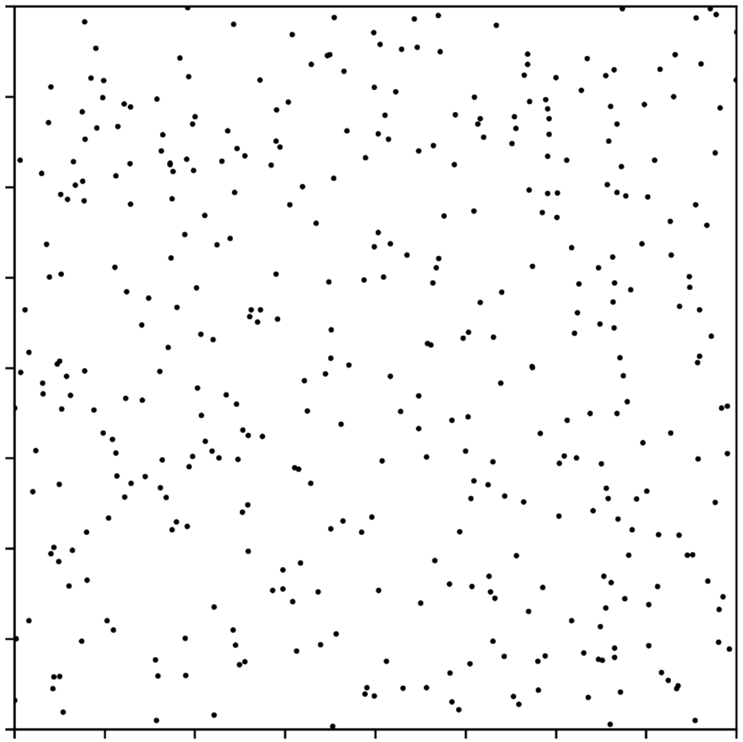
- We consider algorithms A and AD (ours)
- We use $A_{\max} = 4$mm^2 and $D_{\max}$ from 2 to 6mm
- We investigate the performance and solutions obtained with two types of data:
    1. Synthetic data generated within a 20x20mm square region
    2. Real-world medical data already used by Stathonikos et al.

Utrecht University

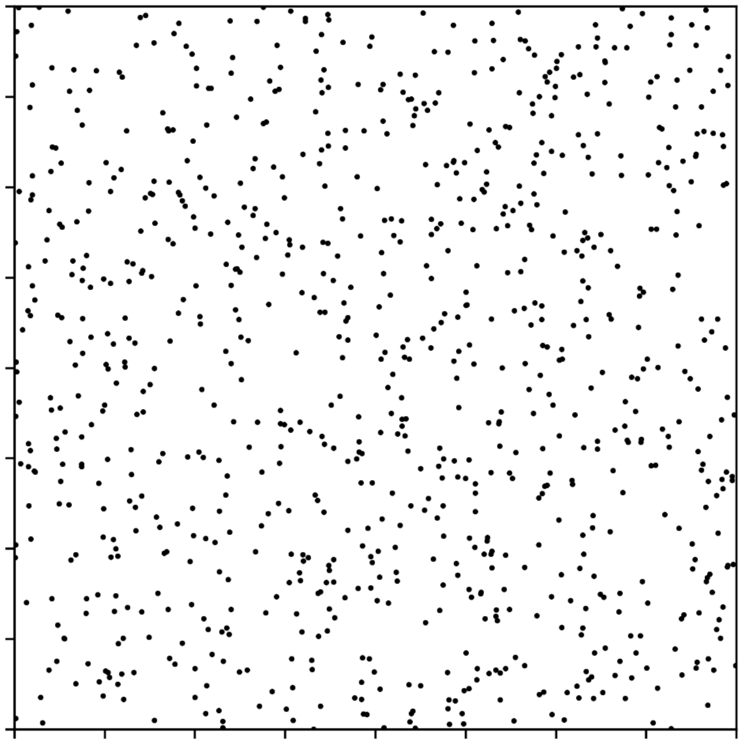
Experimental setup — Uniform data



$n = 100$

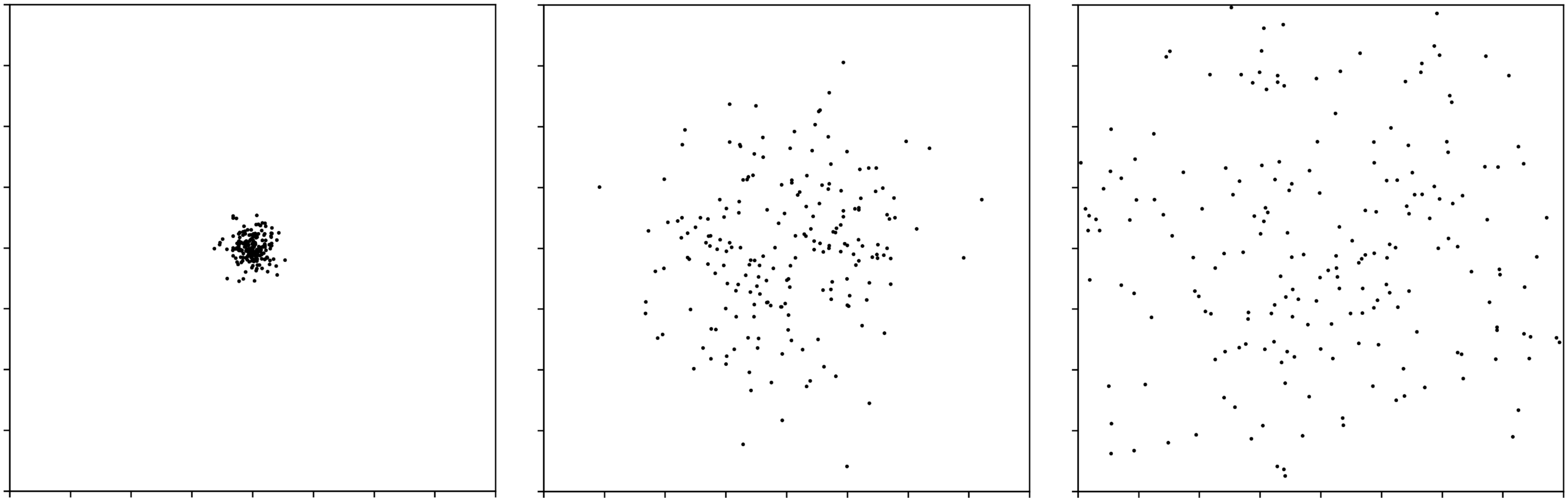


$n = 150$



$n = 200$

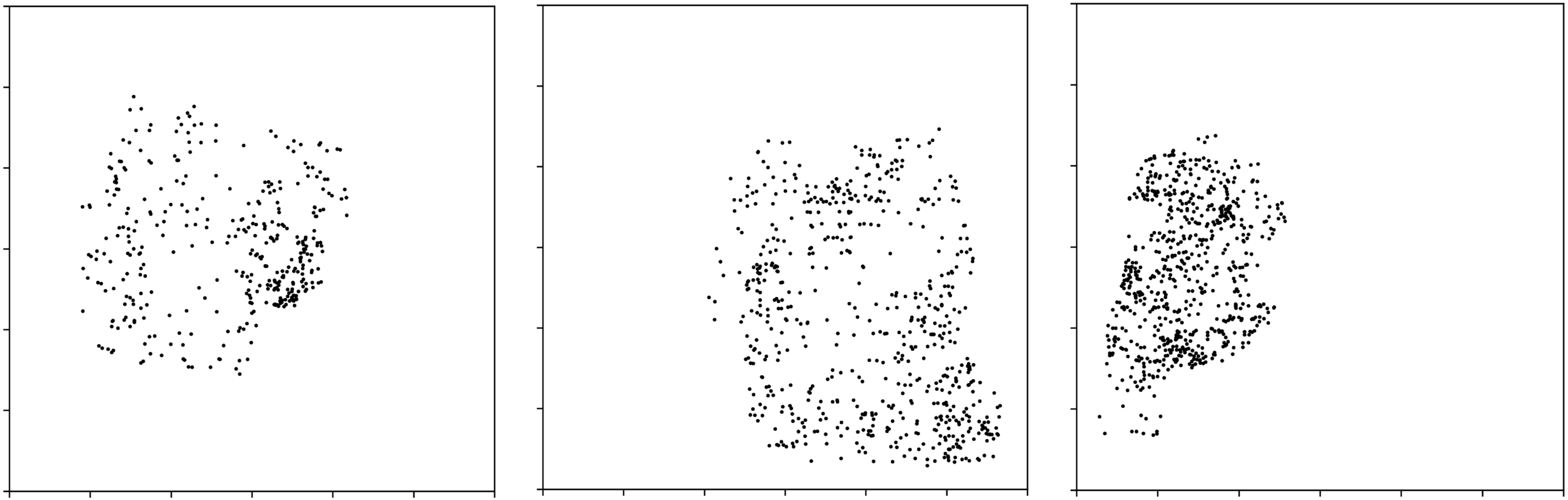
Experimental setup — Gaussian data



$\sigma = 0.5\text{mm}$ $\sigma = 3\text{mm}$ $\sigma = 6.5\text{mm}$

Utrecht University

Experimental setup — Real-world medical data

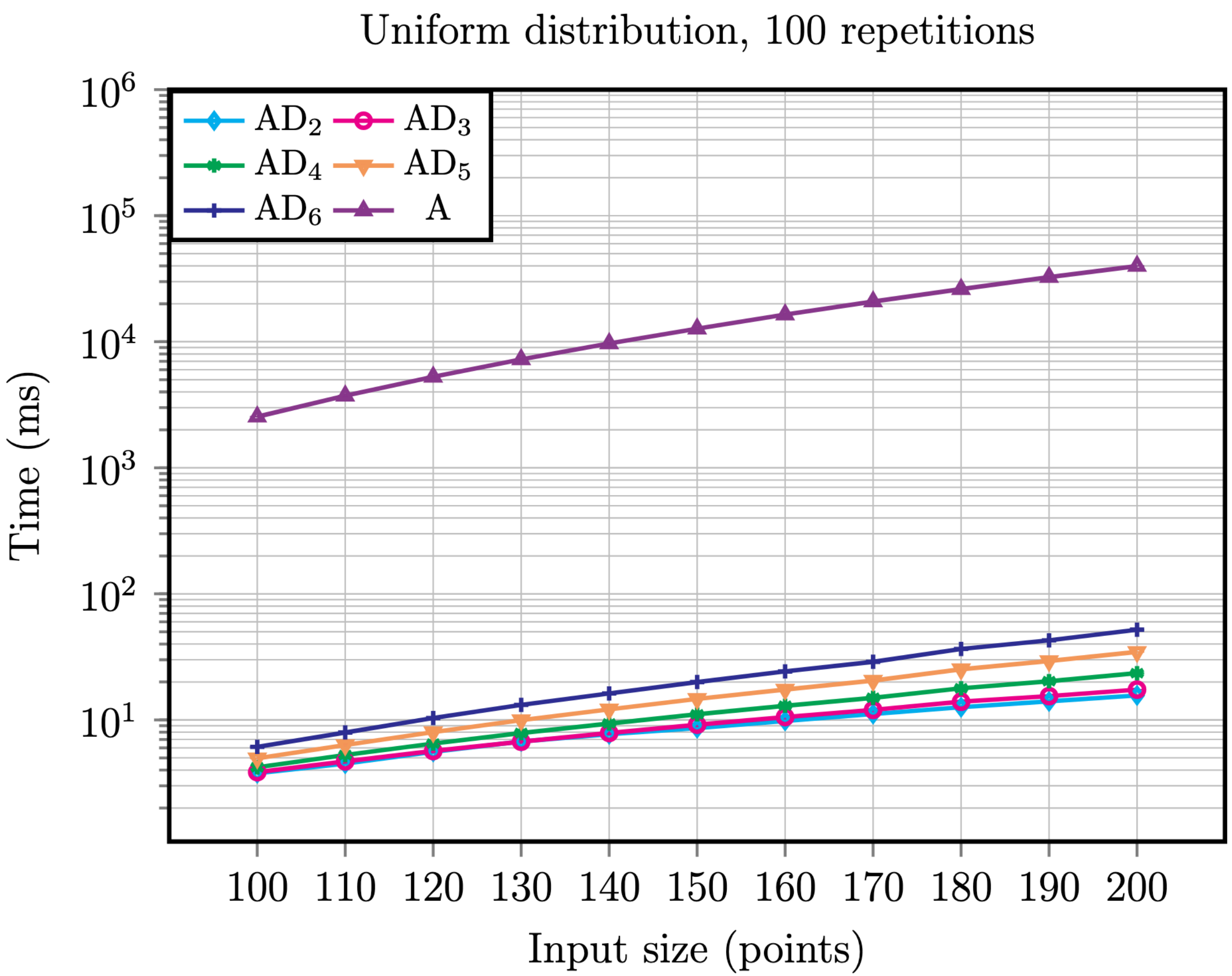


$n = 362$, ANN $= 0.48$mm

$n = 623$, ANN $= 0.31$mm

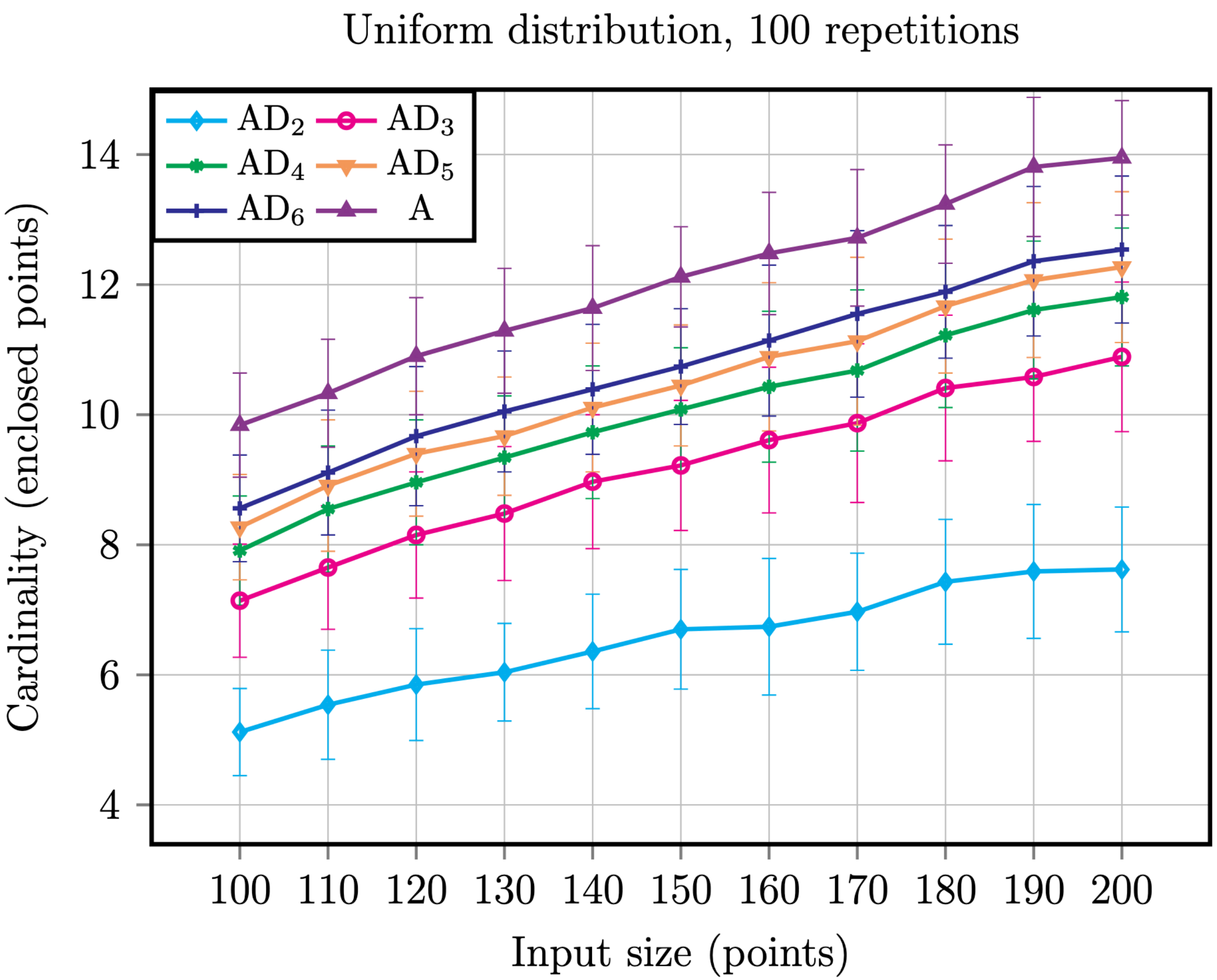
$n = 699$, ANN $= 0.2$mm

Experimental results — Uniform data

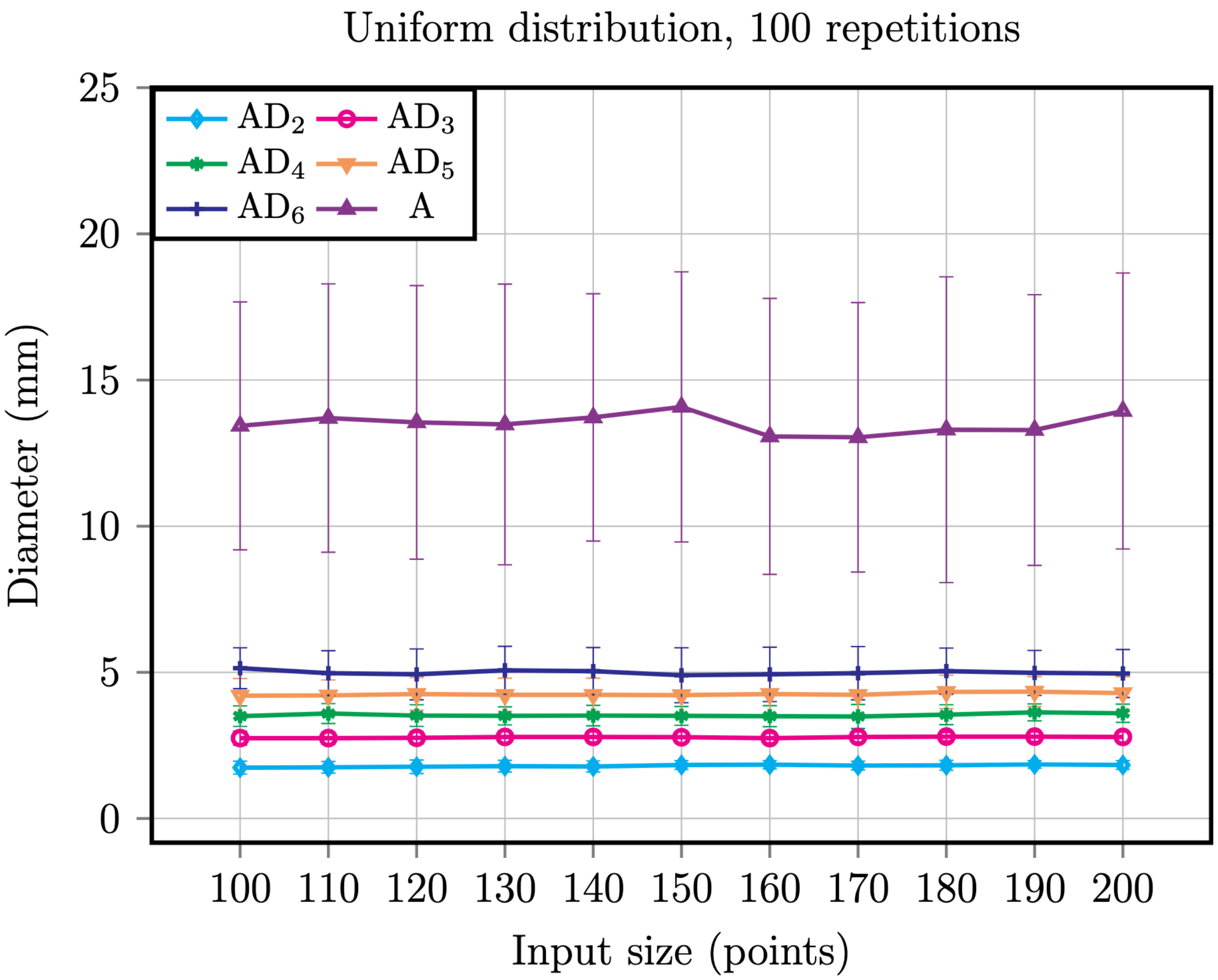


Utrecht University

Experimental results — Uniform data

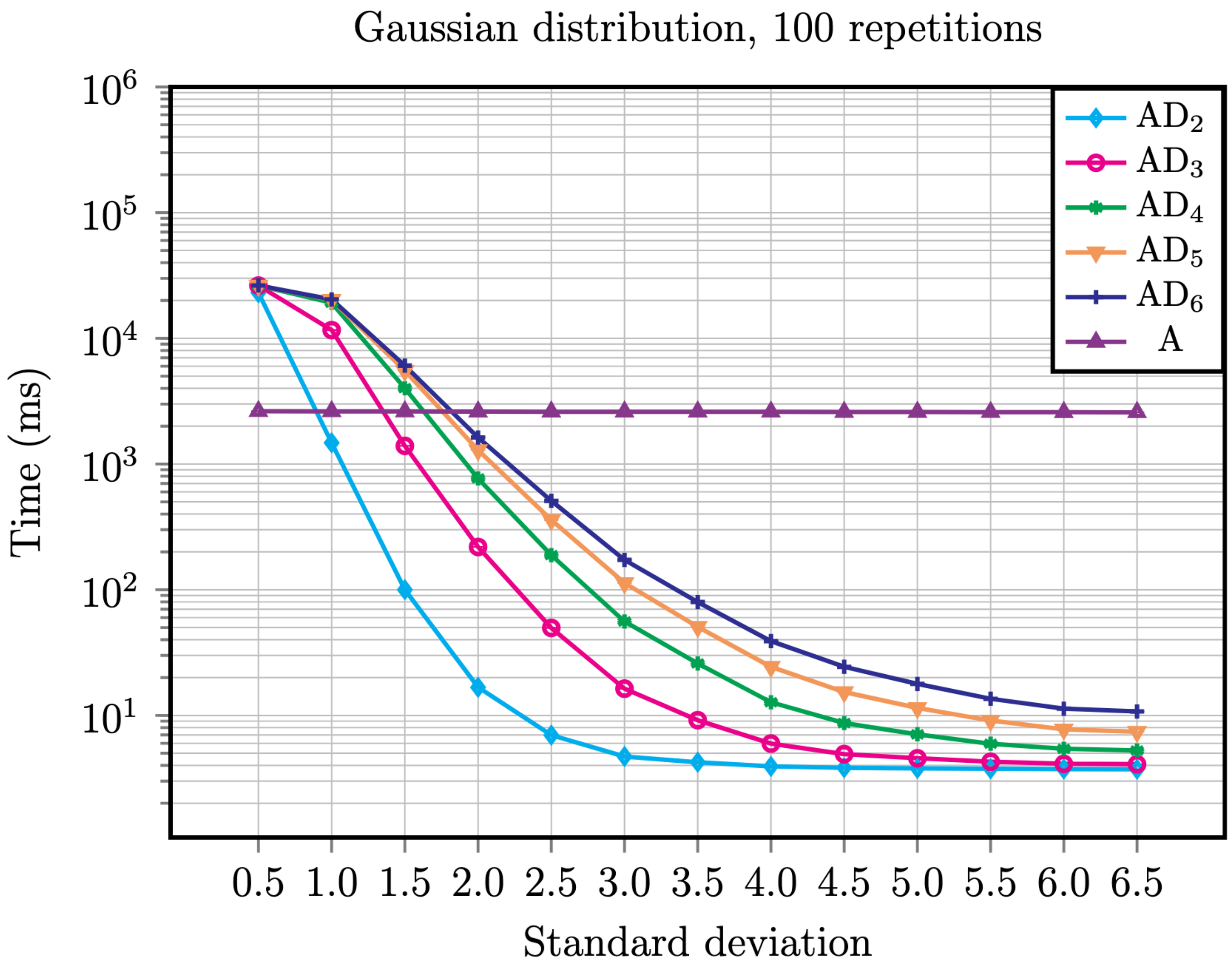


Experimental results — Uniform data

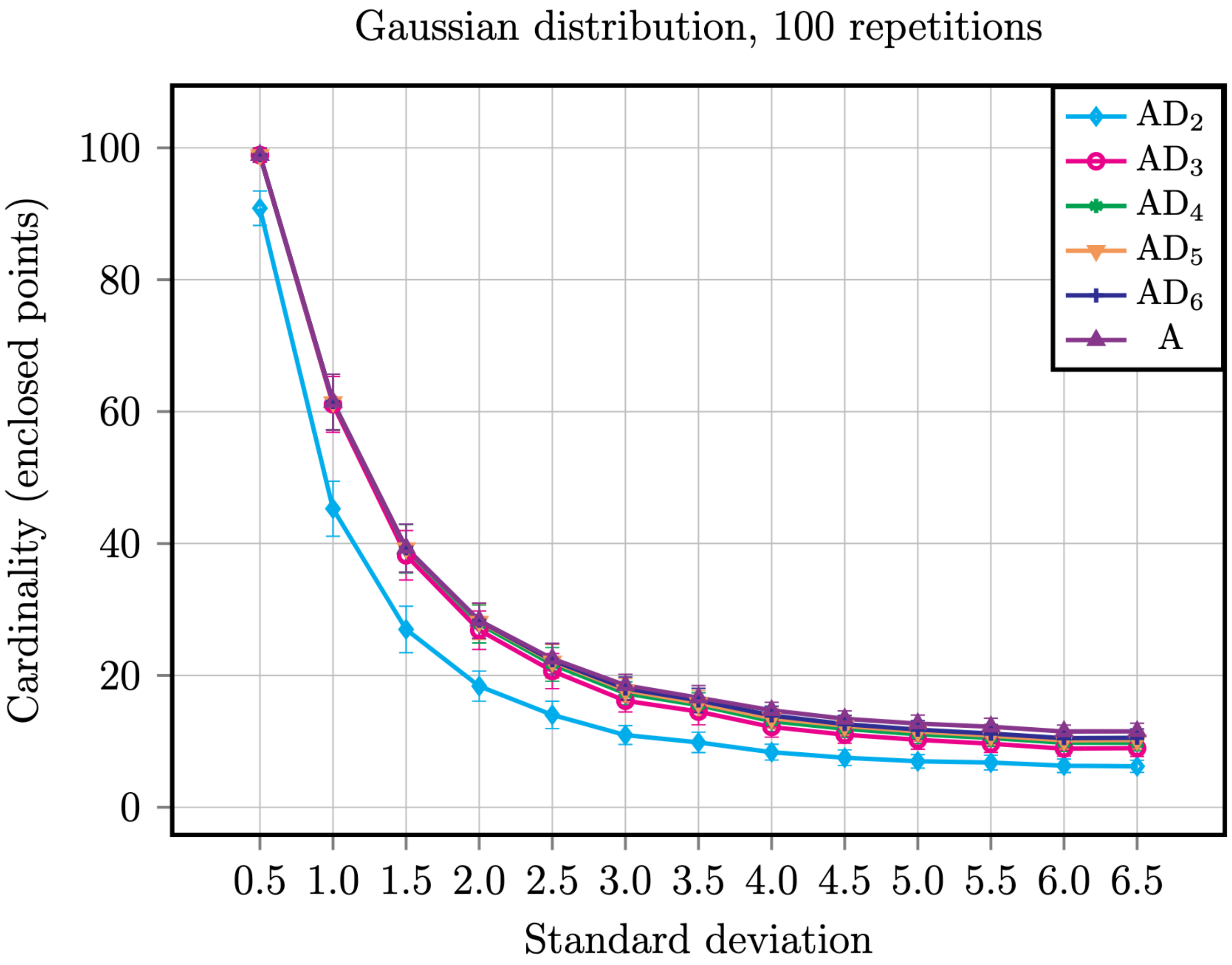


Utrecht University

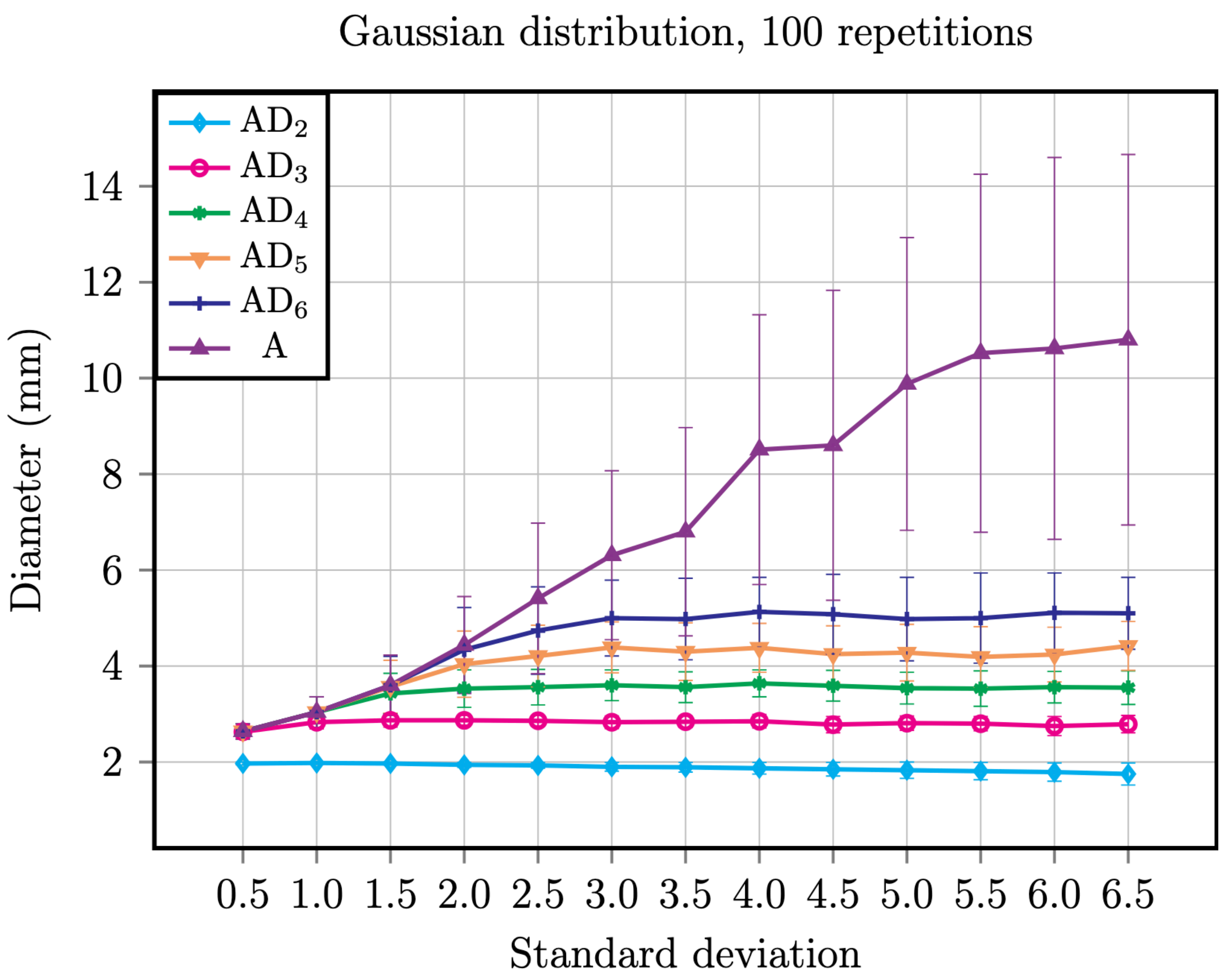
Experimental results — Gaussian data



Experimental results — Gaussian data



Experimental results — Gaussian data



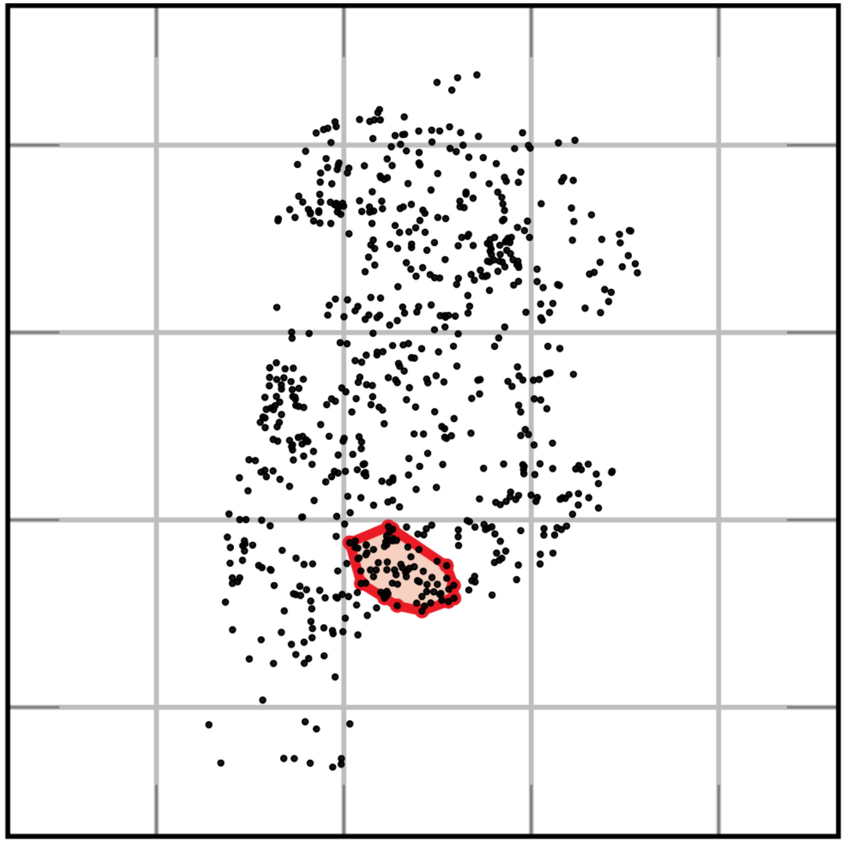
Utrecht University

Experimental results — Real-world medical data

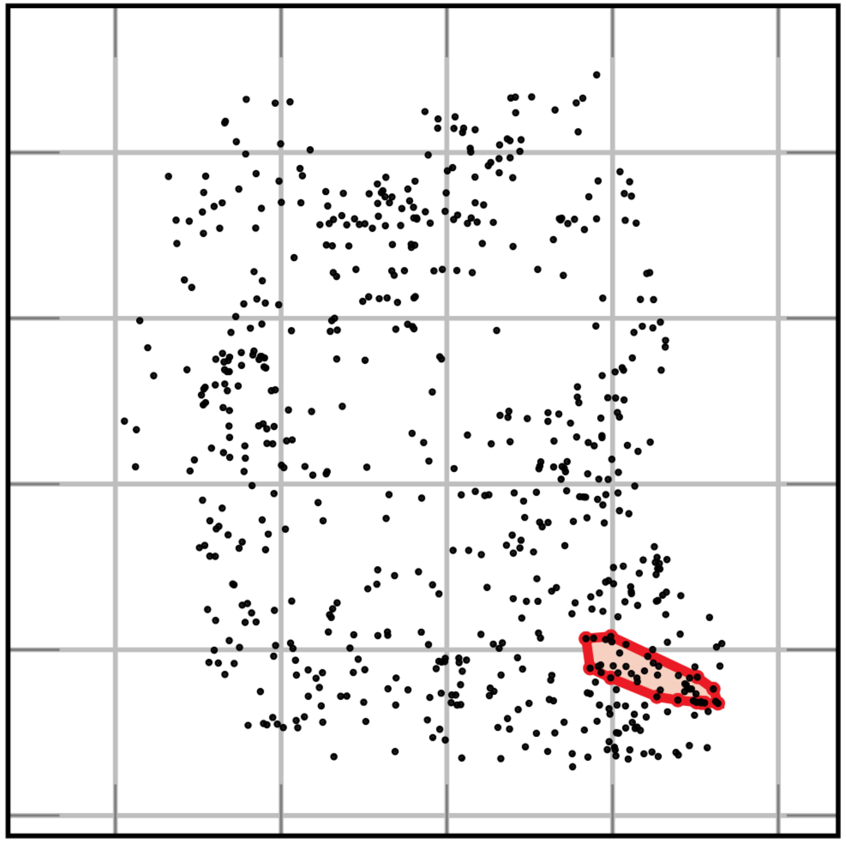
- Points distribution affects performance most

$\mathbf{AD}_4$					
I	time	n	k	area	dia.
1	494.19	699	**76**	3.99	3.15
2	14.7	623	**42**	3.83	3.99
3	29.38	594	**39**	3.93	2.72
4	24.43	526	**42**	3.94	3.1
5	8.83	492	**42**	3.88	2.99
6	21.76	474	**44**	3.88	3.7
7	2.08	412	**31**	3.98	3.85
8	6.19	382	**34**	3.89	3.22
9	37.18	377	**61**	3.91	3.94
10	0.19	362	**28**	3.86	3.49

Experimental results — Real-world medical data



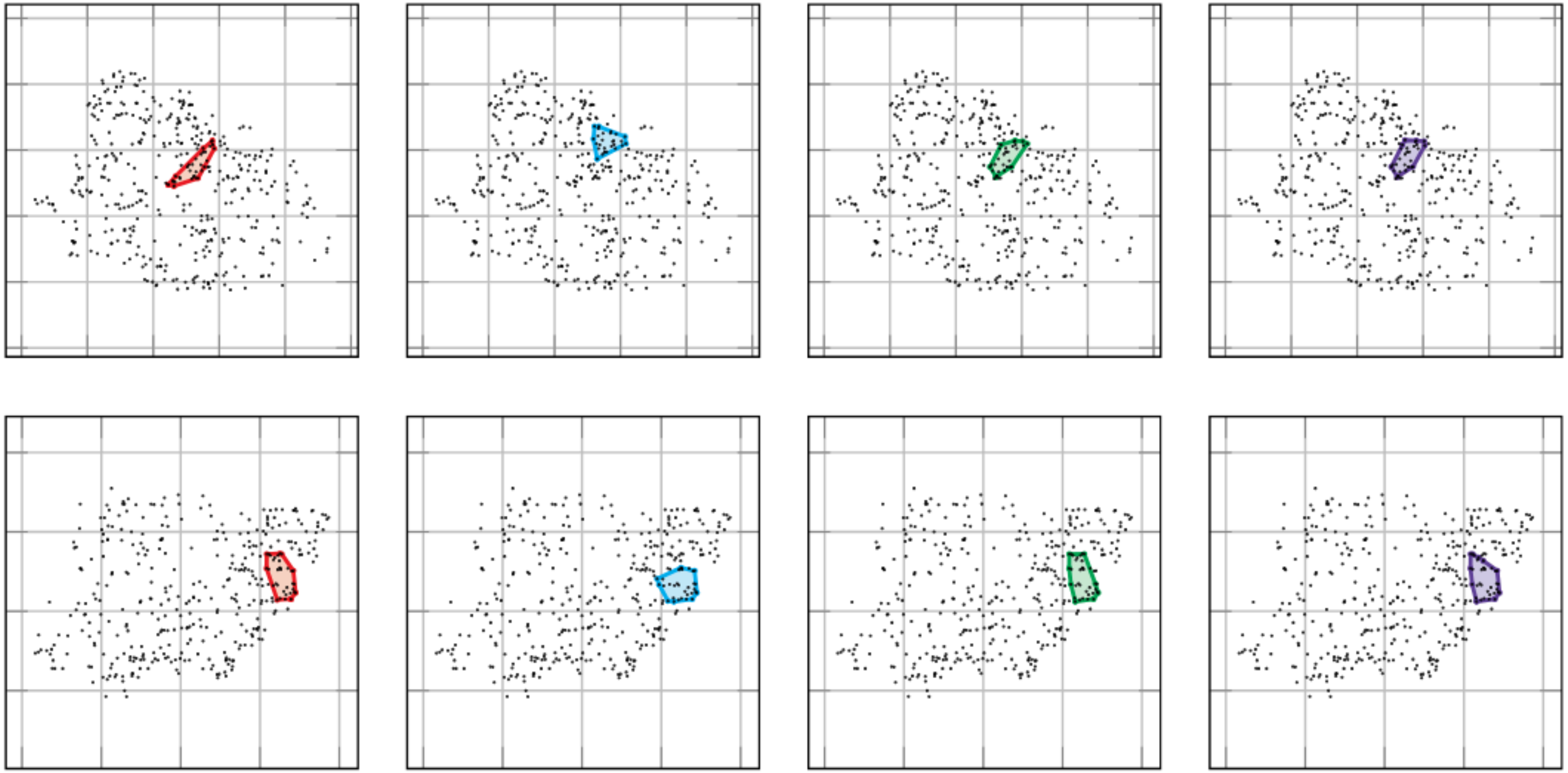
Point set 1



Point set 2

Utrecht University

Experimental results — Real-world medical data



AD_4 $AS(s = 1.98)$ $AS(s = 0.5)$ $A(s = 0.5)$

Conclusion and future work

- Algorithm AD is fast enough to solve problems of real-world size
- Algorithm A tends to find solutions that are too elongated

- Future work:
  1. Improve the runtime of algorithm AD
  2. Design approximation algorithms
  3. Find hotspot based on bounded area and perimeter in polynomial time

Conclusion and future work

- Algorithm AD is fast enough to solve problems of real-world size
- Algorithm A tends to find solutions that are too elongated

- Future work:
  1. Improve the runtime of algorithm AD
  2. Design approximation algorithms
  3. Find hotspot based on bounded area and perimeter in polynomial time

- ***Thank you for your attention!***

Appendix — Experimental results with real-world data (full table)

I	**AD**				**AS** ($s = 1.98$)				**AS** ($s = 0.5$)				**A** ($s = 0.5$)			
	time	k	area	dia.	time	$\hat{k}$	area	dia.	time	$\hat{k}$	area	dia.	time	$\hat{k}$	area	dia.
0	494.19	**76**	3.99	3.15	5.24	68	3.42	2.89	72.83	70	3.74	2.88	13.7	74	3.9	3.07
1	14.7	**42**	3.83	3.99	17.15	36	3.84	2.74	274.82	39	3.97	2.9	2.84	40	3.94	2.87
2	29.38	**39**	3.93	2.72	19.89	33	3.99	2.85	218.36	**39**	3.98	2.66	3.66	**39**	3.93	2.72
3	24.43	**42**	3.94	3.1	19.12	34	3.99	2.71	263.07	39	3.97	3.15	3.17	40	3.88	3.1
4	8.83	**42**	3.88	2.99	13.83	38	3.87	2.82	229.33	40	3.99	2.83	2.08	**42**	3.98	2.88
5	21.76	**44**	3.88	3.7	9.16	37	3.82	3.31	141.13	39	3.76	3.17	4.51	39	3.82	3.01
6	2.08	**31**	3.98	3.85	9.32	27	3.72	2.73	161.07	28	4.0	3.45	1.63	28	3.94	3.38
7	6.19	**34**	3.89	3.22	13.66	30	3.94	2.65	174.71	32	3.92	3.22	1.82	**34**	3.92	3.22
8	37.18	**61**	3.91	3.94	9.66	57	3.86	2.67	77.41	59	3.97	2.72	2.58	60	3.92	3.01
9	0.19	**28**	3.86	3.49	4.83	23	3.97	3.41	52.02	27	3.97	2.74	1.47	27	3.97	2.74