Data Structures and Algorithms for tractable statistical mining of high dimensional data

Anna Goldenberg
Auton Lab

Ting Liu
Auton Lab

Andrew Moore
Auton Lab

Daniel Neill
Auton Lab

Papers, example datasets, software available from www.autonlab.org

The Auton Lab
School of Computer Science
Carnegie Mellon University
www.autonlab.org

The Auton Lab

Faculty:	Andrew Moore (Prof), Jeff Schneider (Research Scientist), Artur Dubrawski (Systems Scientist)
Postdoctoral Fellows:	Brigham Anderson, Alexander Gray, Paul Komarek, Dan Pelleg, Josep Roure
Graduate Students:	Brent Bryan, Kaustav Das, Khalid El-Arini, Anna Goldenberg, Jeremy Kubica, Ting Liu, Daniel Neill, Jens Neilsen, Sajid Siddiqi, Purna Sarkar, Ajit Singh
Head of Software Development:	Jeanie Komarek
Programmers:	Karen Chen, Patrick Choi, Adam Goode, Pat Gunn, Joey Liang, John Ostlund, Robin Sabhnani, Rahul Sankathar
Executive Assistant:	Kristen Schrauder
Head of Sys. Admin:	Mike Baysek, Jacob Joseph
Undergraduate and Masters Interns:	Kenny Daniel, Sandy Hsu, Dongryeol Lee, Jennifer Lee, Avilay Parekh, Chris Rotella, Jonathan Terleski
Recent Alumni:	Drew Bagnell (RI faculty), Scott Davies (Google), David Cohn (Google), Geoff Gordon (CMU), Paul Hsiung (USC), Marina Meila (U. Washington), Remi Munos (Ecole Polytechnique), Malcolm Strens (Qinetiq)

Outline

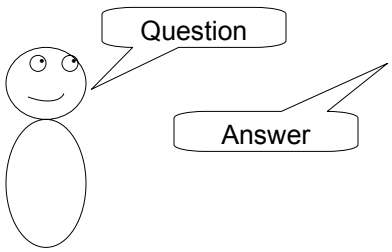
- Cached Sufficient Statistics
- Ball Trees Refresher
- K-nearest-neighbor classification (exploiting the question part one)
- Non-parametric classification

- Biosurveillance and Epidemiology
- Scan Statistics (exploiting the question part two)

- Bayesian Network Learning
- Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

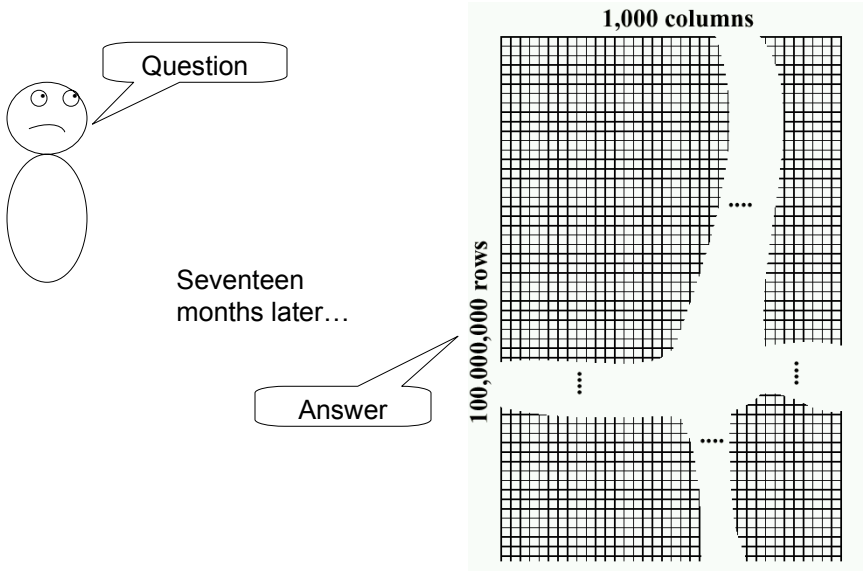
- Sensible conclusion
- Flaky conclusion

Data Analysis: The old days

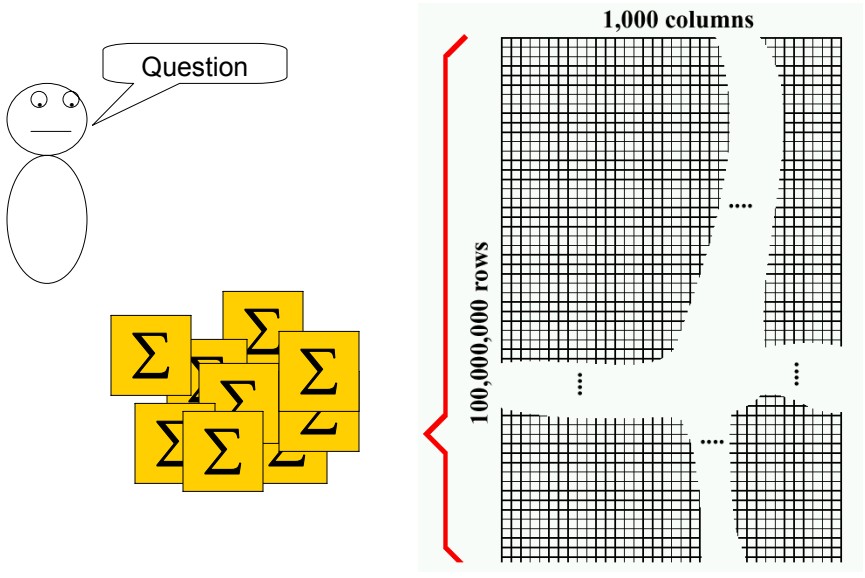


Size	Ellipticity	Color
23	0.96	Red
33	0.55	Red
36		Green
40		
20		
48		

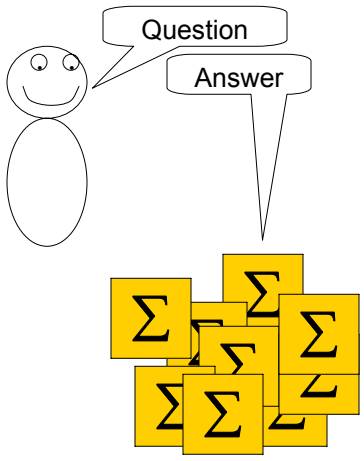
Data Analysis: The new days

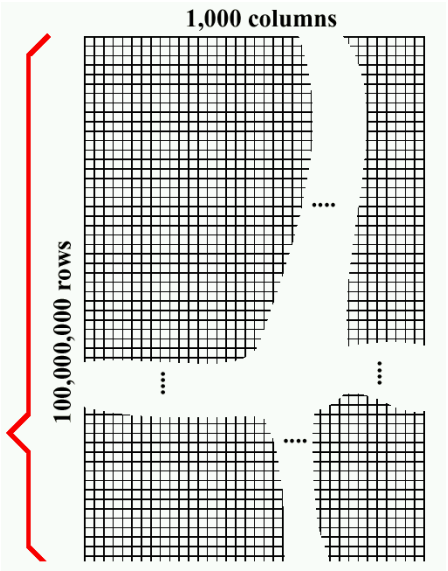


Cached Sufficient Statistics



Cached Sufficient Statistics



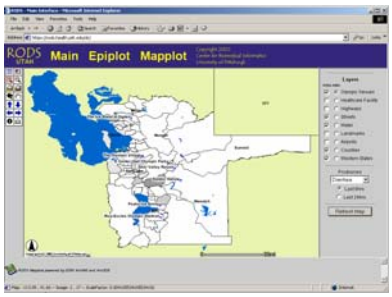


1997	1998	1999	2000	2001	2002	2003	2004
M&M Mars Line control Adrenaline (NOX mini-malization)	Kodak (Image sta-bilization) Digital Equipment (pregnan-cy monit-oring)	M&M Mars (manufac-turing) NASA/ NSF (Astrophysics mining) 3M Textile tension control	Caterpillar (Spare parts) US Army (biotoxin detection) M&M Mars Scheduling with uncer-tainty 3M (Adhesive design)	DigitalMC (Music tastes) Caterpillar (emissions) SmartMoney (anomalies) Unilever (Brand Management) Phillips Petroleum (work-force optimization) Cellomics (screened anomaly detection)	Biometrics company (health monitor) Boeing (intrusion) Masterfoods (new product development) Cellomics (pro-teomics screen) ABB (Circuit-breaker supply chain) SwissAir (Flight delays) 3M (secret) Washington Public Hospital System (ER delays) Unilever (targeted marketing)	NASA (National Virtual Observatory) NSF (astrostatistics software) DARPA (national disease monitor) Masterfoods (bio-chemistry) Pfizer (High-throughput screen) Caterpillar Inc. (Self-optimizing Engines) Beverage Company (Ingredients/Manufacturing/Marketing/ Sales Bayes Net) Transform Pharma (massive autonomous experiment design) Census Bureau (privacy protection) Psychogenics Inc: Effects of psychotropic drugs on rats	NSF (astrostatistics software) Masterfoods (bio-chemistry) State of PA (National Disease Monitor [with Mike Wagner of U. Pitt]) State of PA (Anti Cancer [collaboration with CMU Biology] DARPA (detecting patterns in links) Other Government Departments (identifying dangerous people, potential collaborators, and aliases) Other Government Departments (detecting a class of clusters) Other Pharma Research Co. Life Science specific data mining United States Department of Agriculture: Early warning system for food terrorism NSF Biosurveillance Algorithms

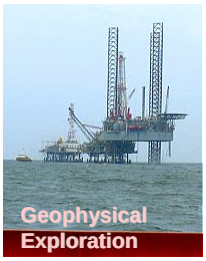
Auton/SPR Deployments

Our 7 biggest applications in 2005

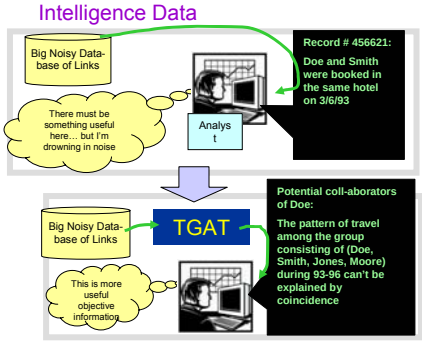
Biomedical Security (with Mike Wagner, University of Pittsburgh)



Autonomous self-tweaking engines



Intelligence Data



Current Sponsors

- National Science Foundation (NSF)
- NASA
- Defense Advanced Research Projects Agency (DARPA)
- Central Intelligence Agency (CIA)
- Department of Homeland Security (DHS)
- Homeland Security Advanced Research Projects Agency (HSARPA)
- United States Department of Agriculture (USDA)

Federal

- Health Canada
- State of Pennsylvania

State

- Pfizer Inc.
- Caterpillar Corporation
- British Petroleum

Large industrial

- Psychogenics Corporation
- Transform Pharmaceuticals

Small industrial

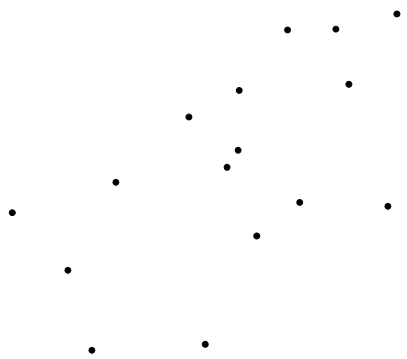
Outline

- Cached Sufficient Statistics
- Ball Trees Refresher
- K-nearest-neighbor classification (exploiting the question part one)
- Non-parametric classification

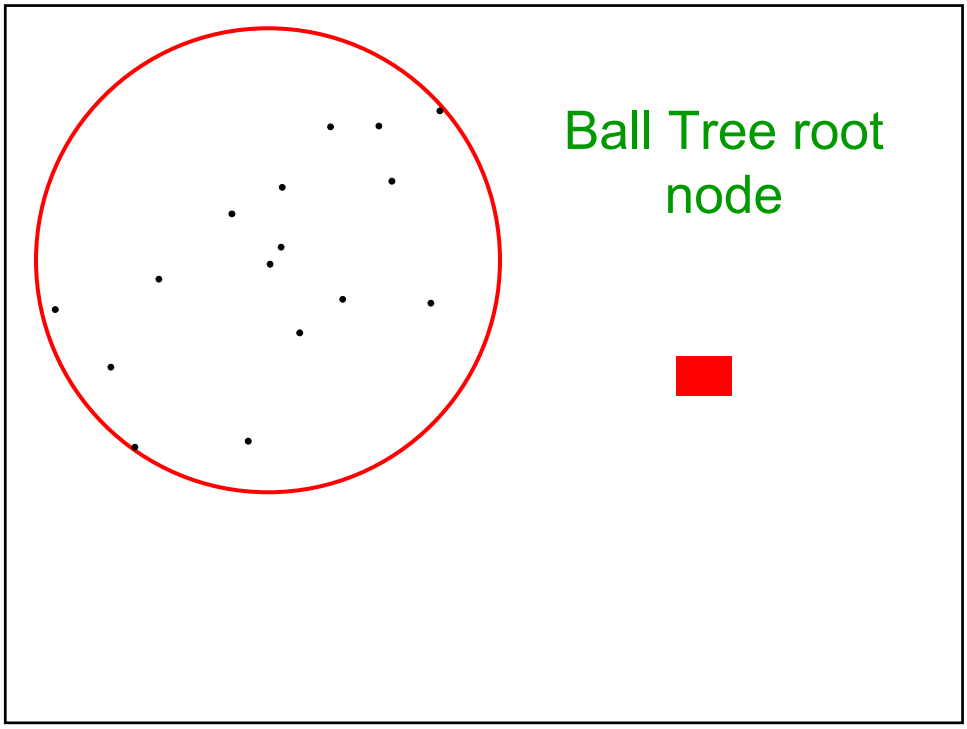
- Biosurveillance and Epidemiology
- Scan Statistics (exploiting the question part two)

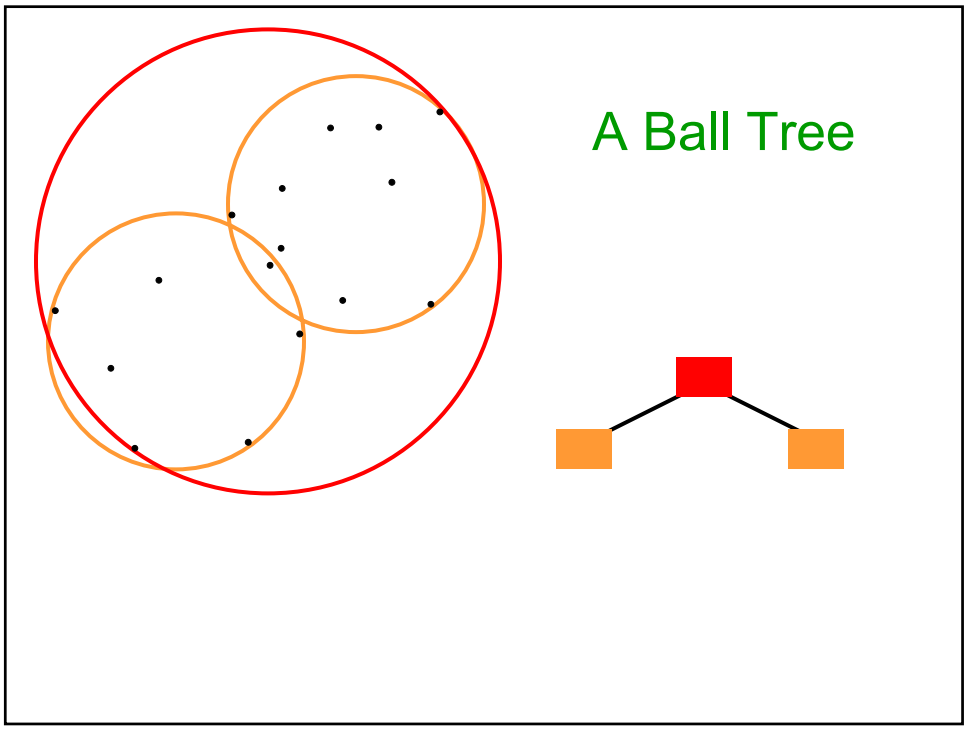
- Bayesian Network Learning
- Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

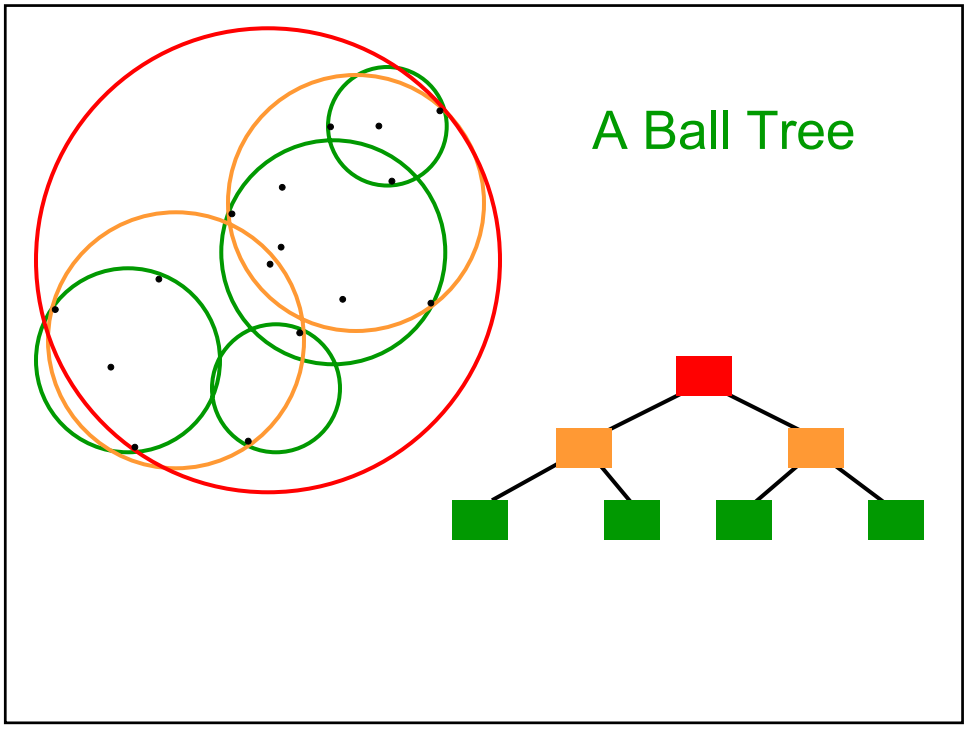
- Sensible conclusion
- Flaky conclusion

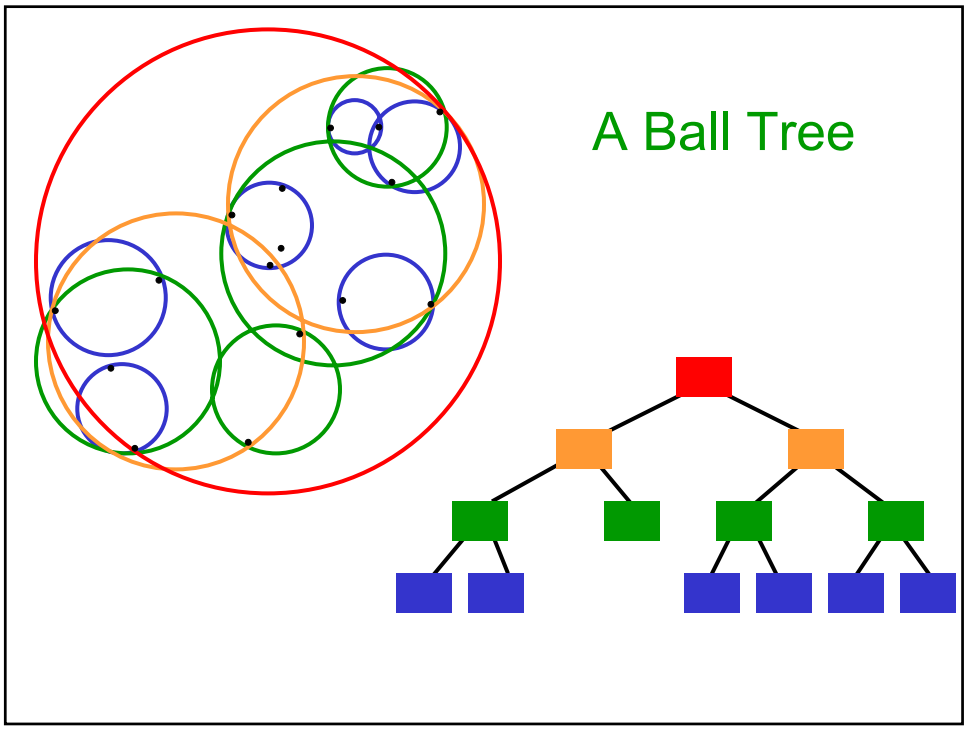


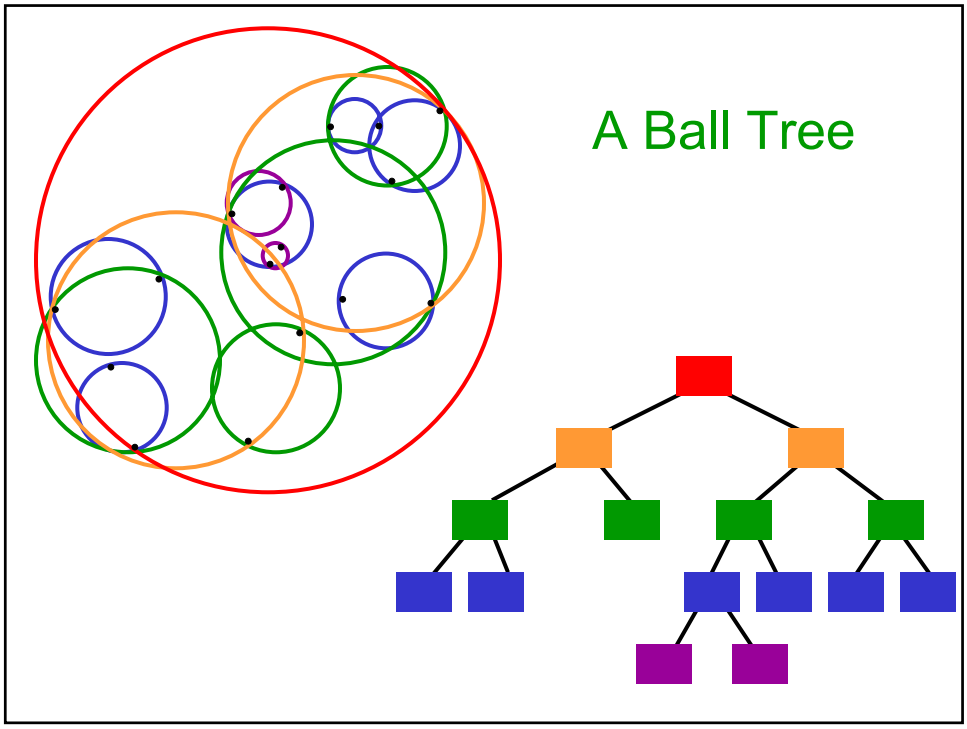
A Set of Points in a metric space

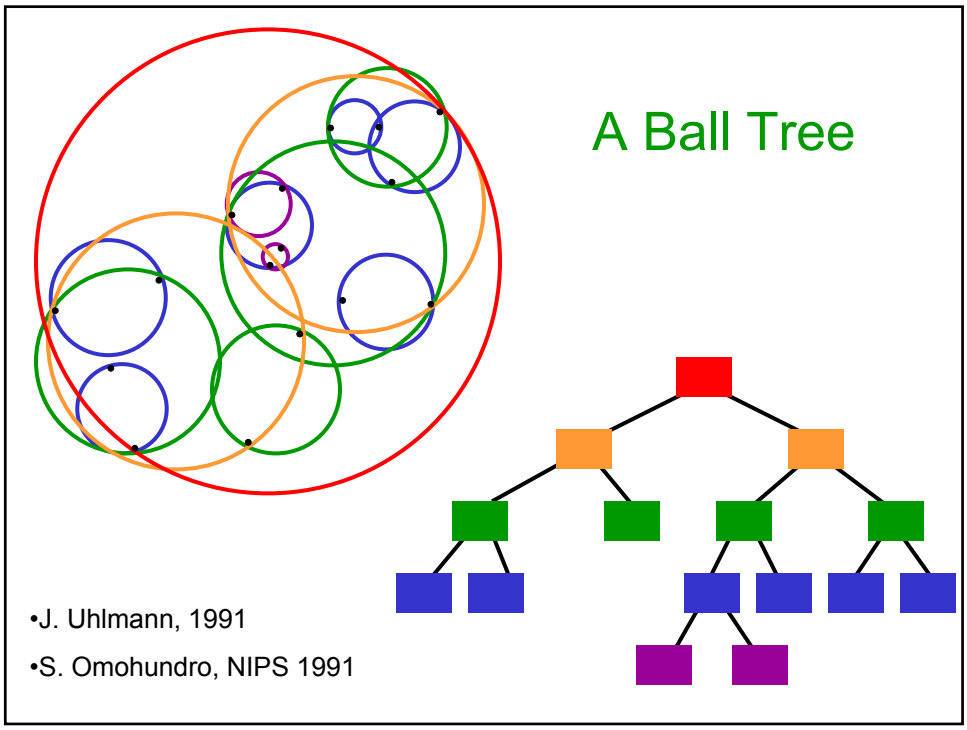

Ball Tree root
node


A Ball Tree


A Ball Tree


A Ball Tree


A Ball Tree

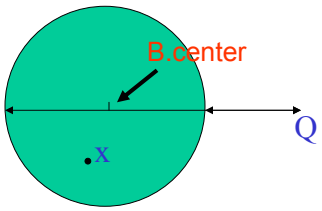

A Ball Tree
•J. Uhlmann, 1991
•S. Omohundro, NIPS 1991

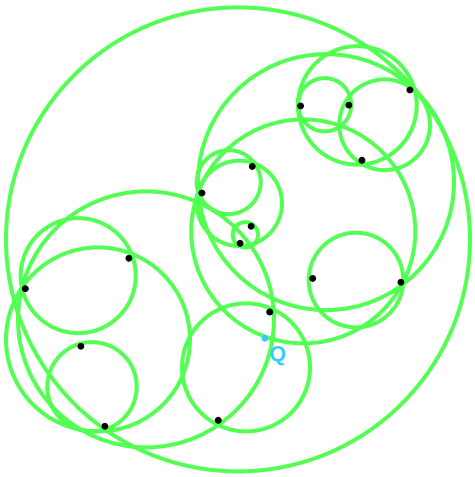
Ball-trees: properties

Let *Q* be any query point and let *x* be a point inside ball *B*

$|x-Q| \geq |Q - B.center| - B.radius$

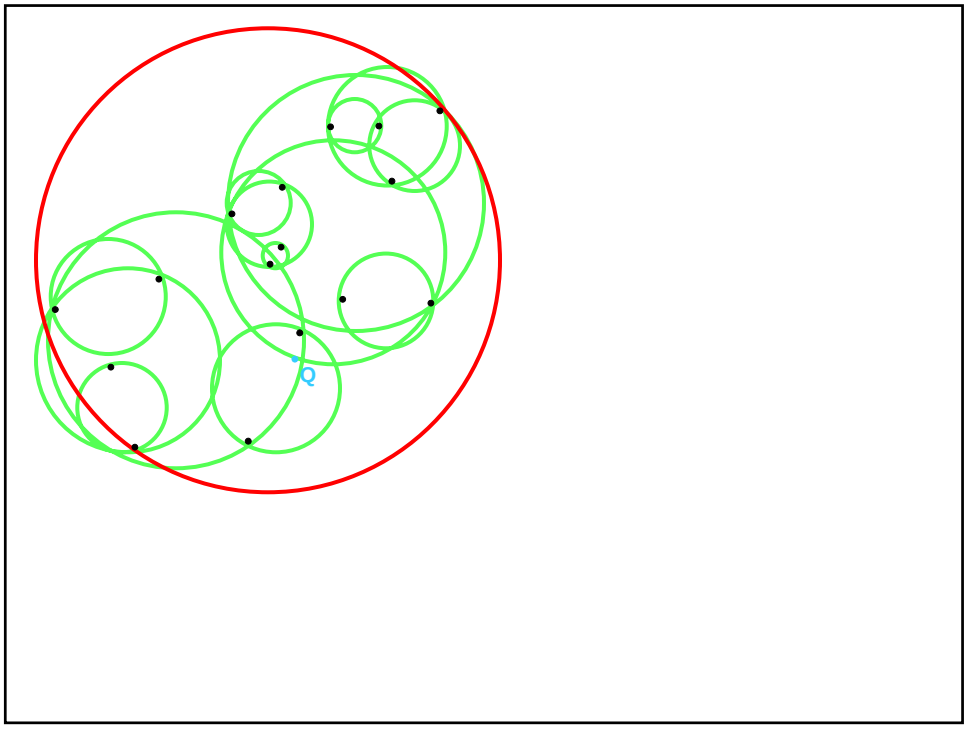
$|x-Q| \leq |Q - B.center| + B.radius$

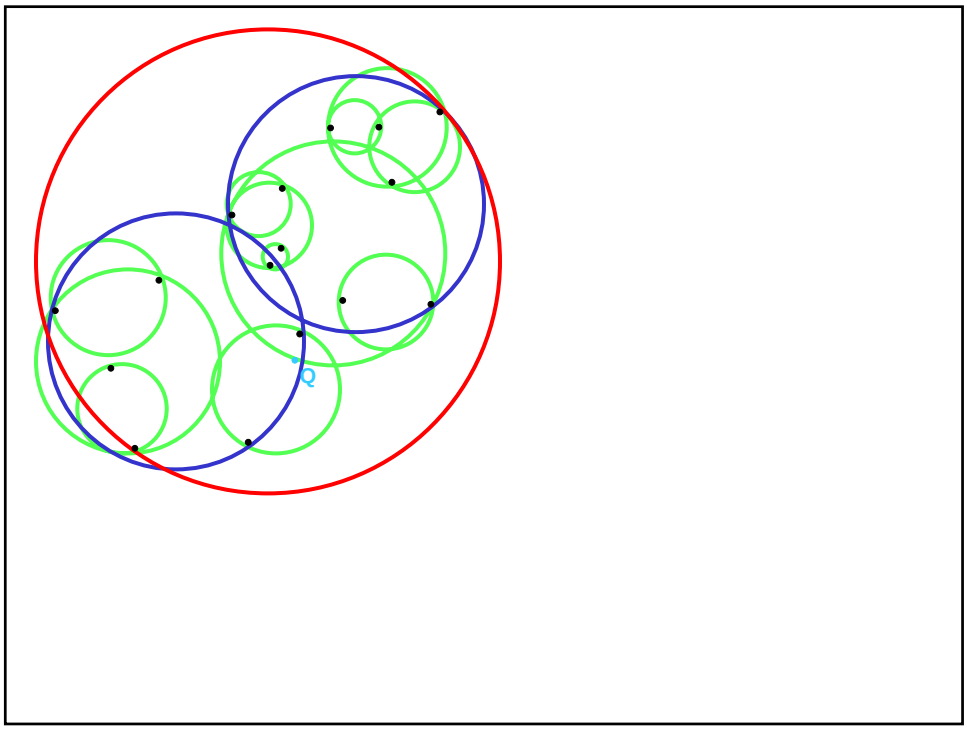


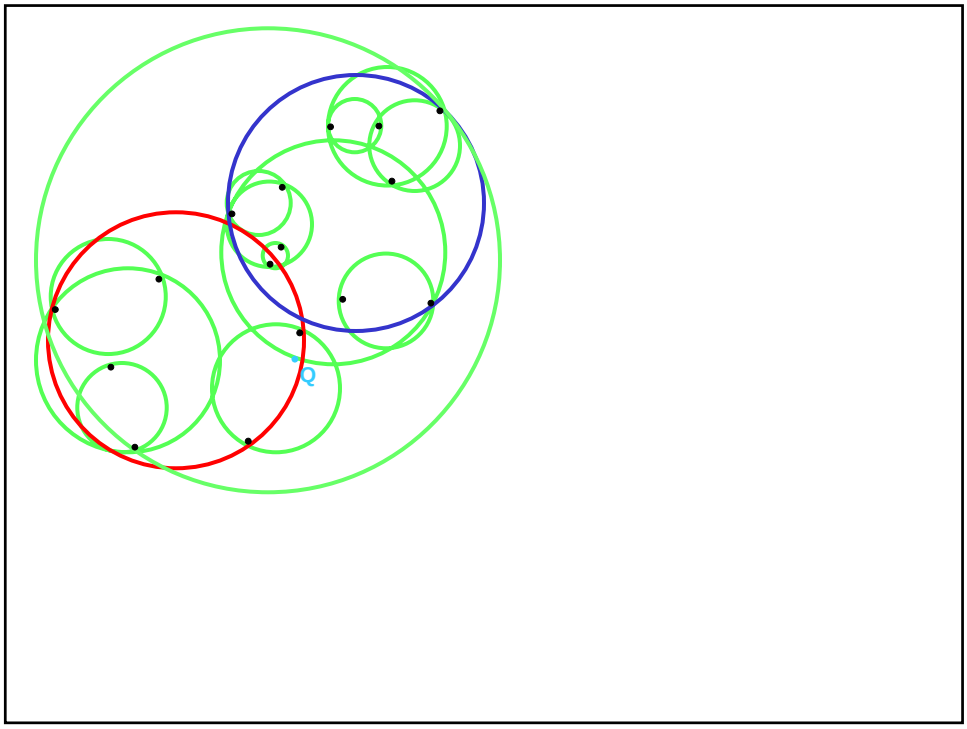


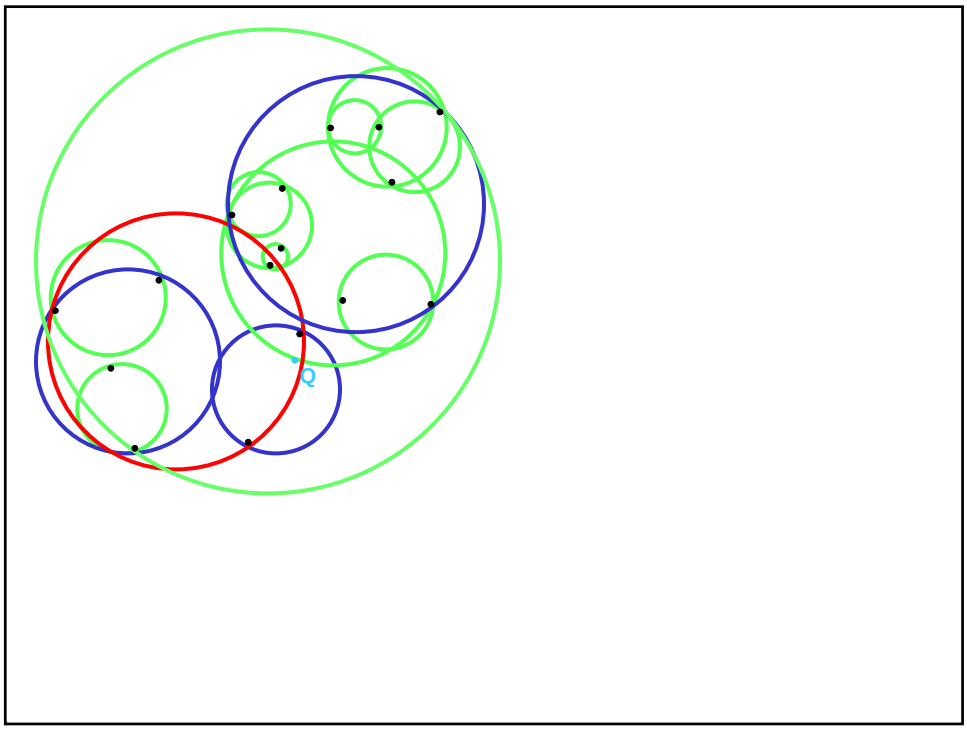
Goal: Find out the 2-nearest neighbors of Q.

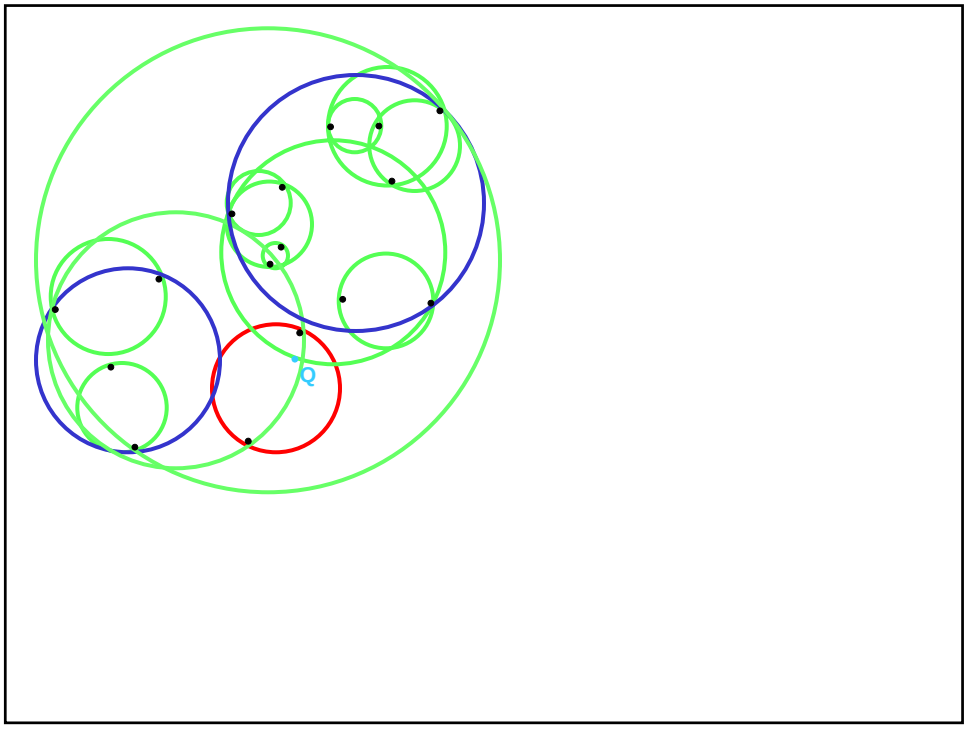
- J. Uhlmann, 1991
- S. Omohundro, NIPS 1991

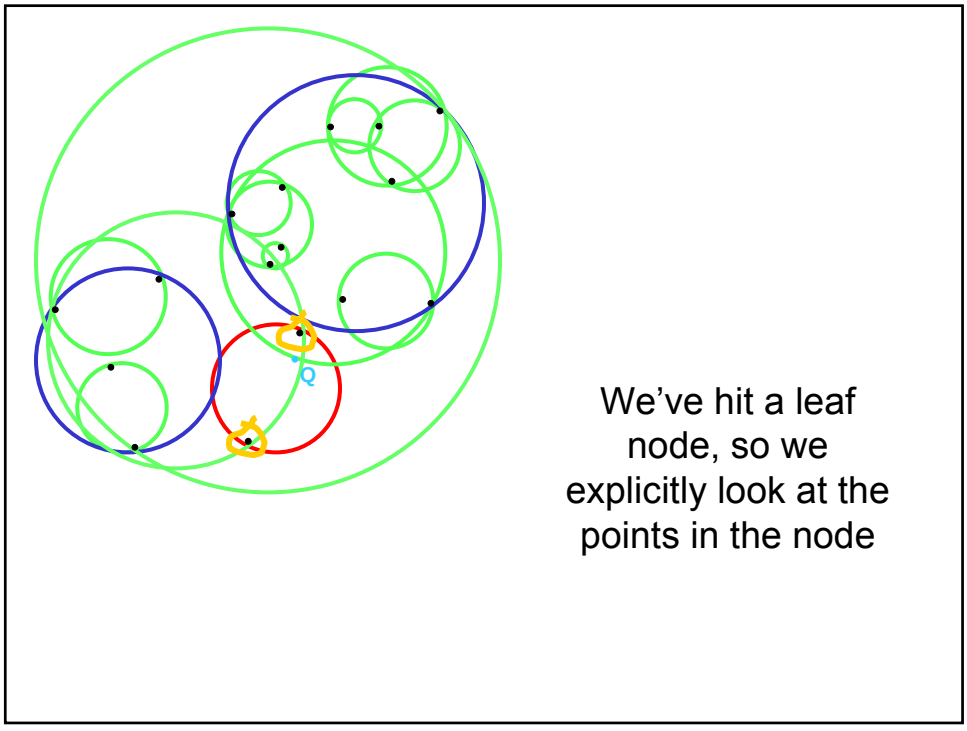

Q


Q

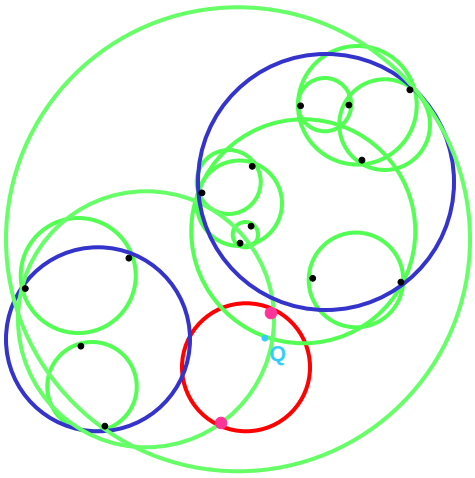

Q


Q



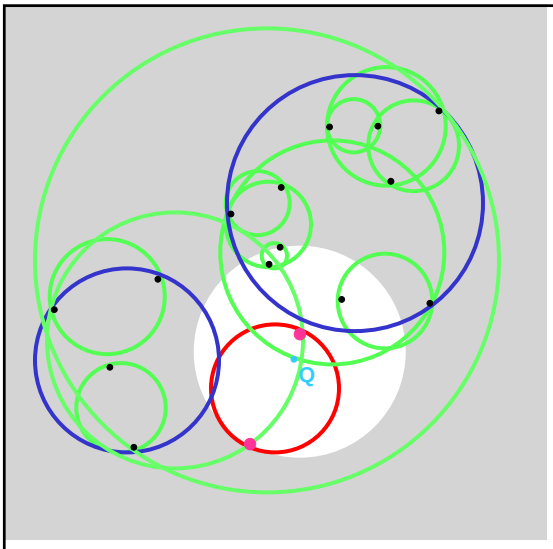


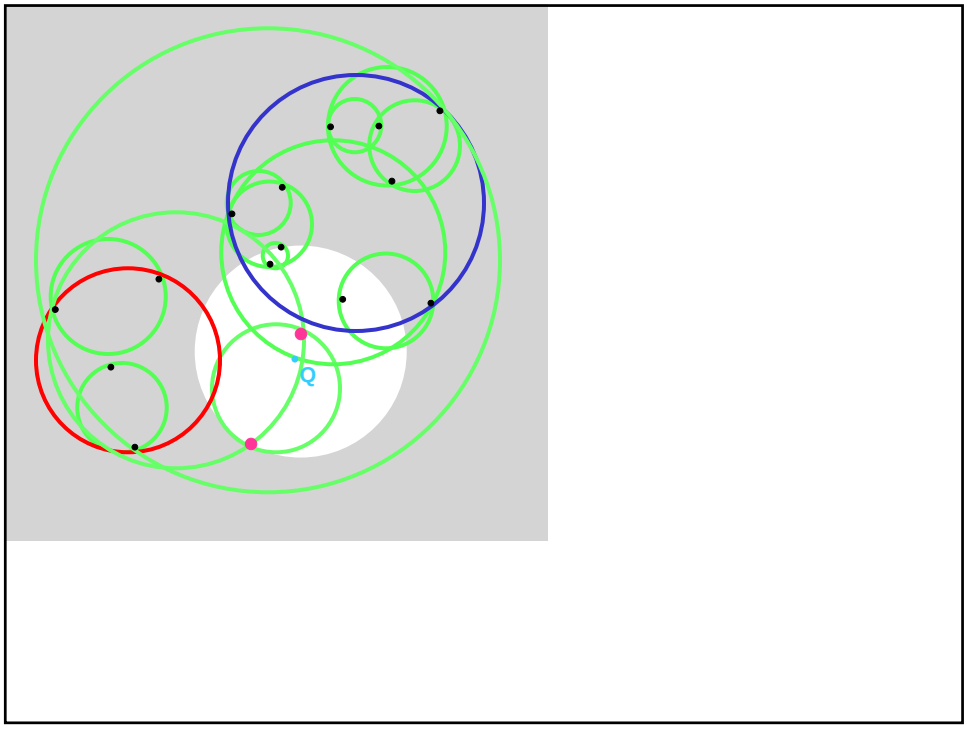
We've hit a leaf node, so we explicitly look at the points in the node

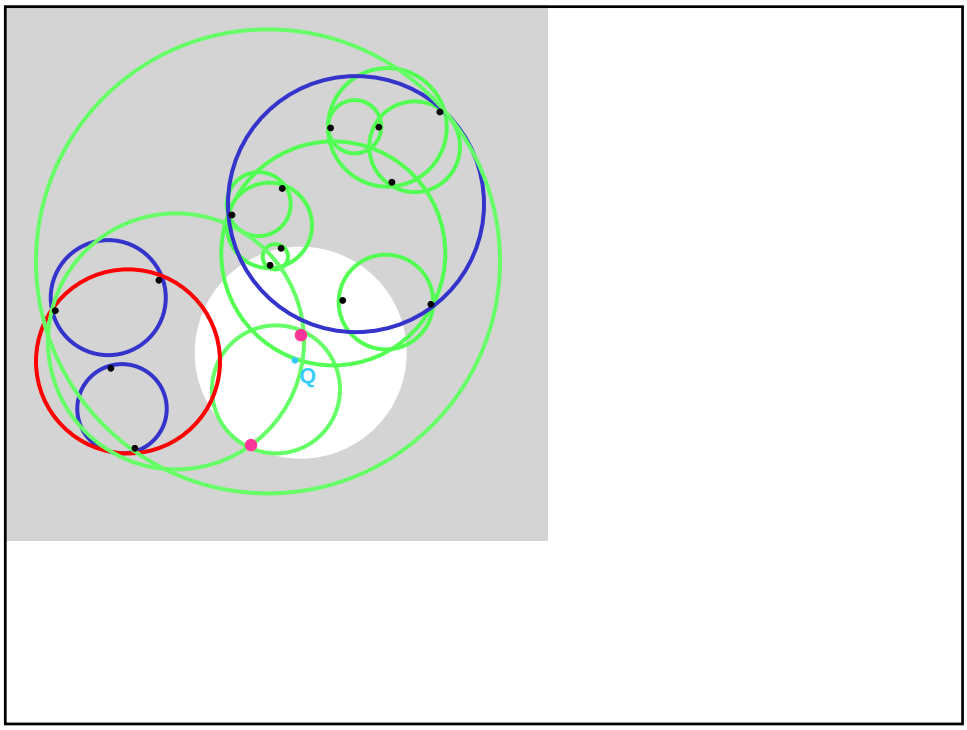


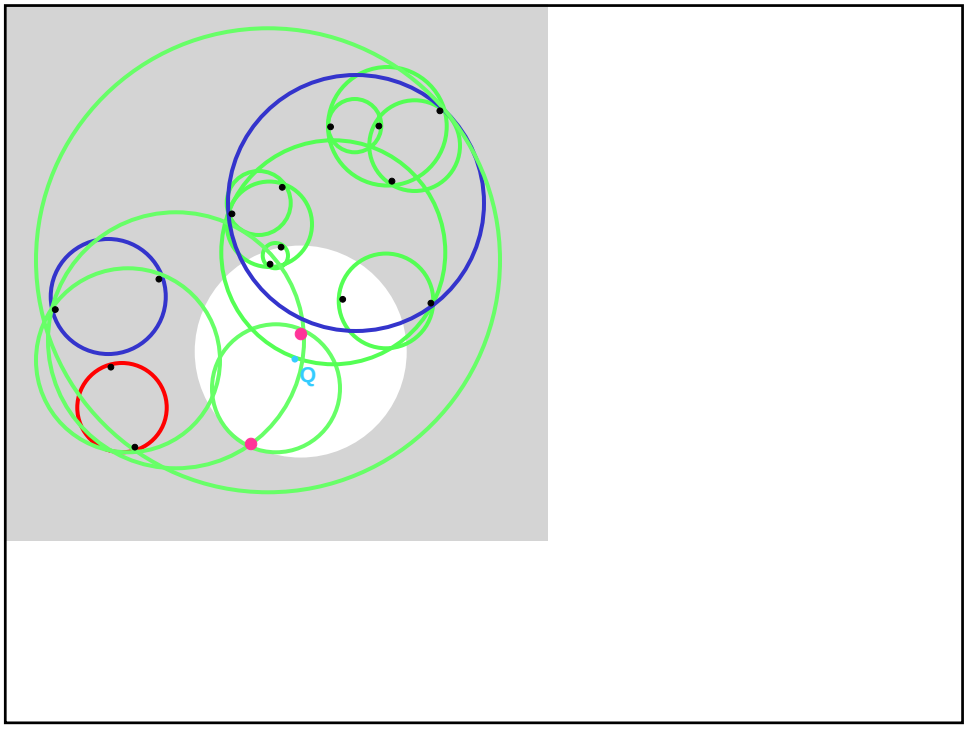
Two nearest neighbors found so far are in pink

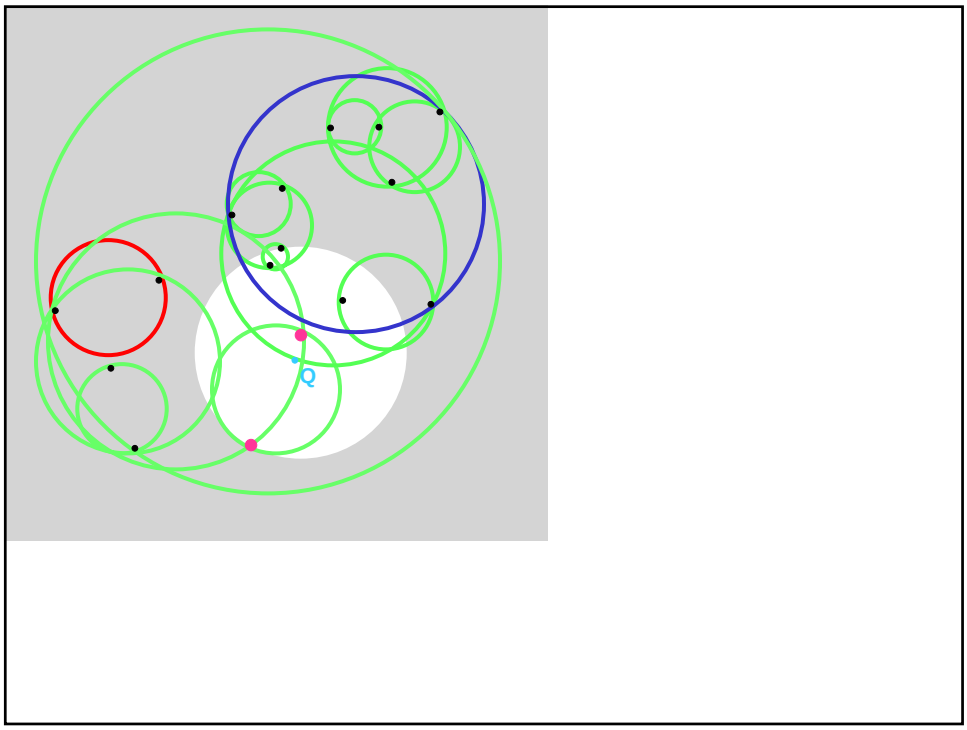
(remember we have yet to search the blue balls)

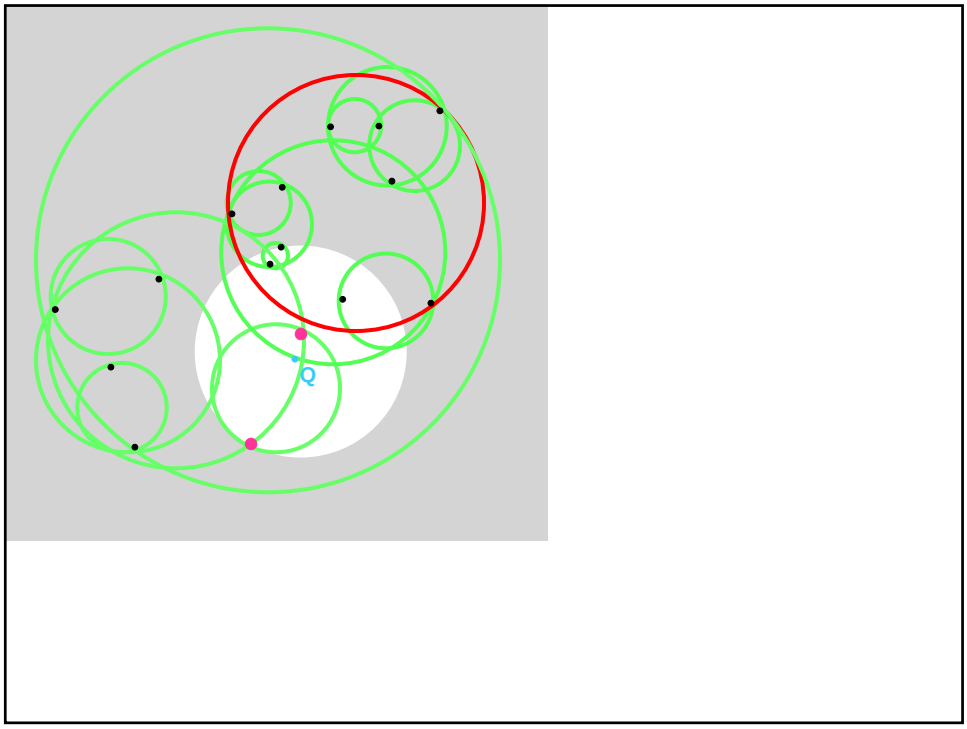


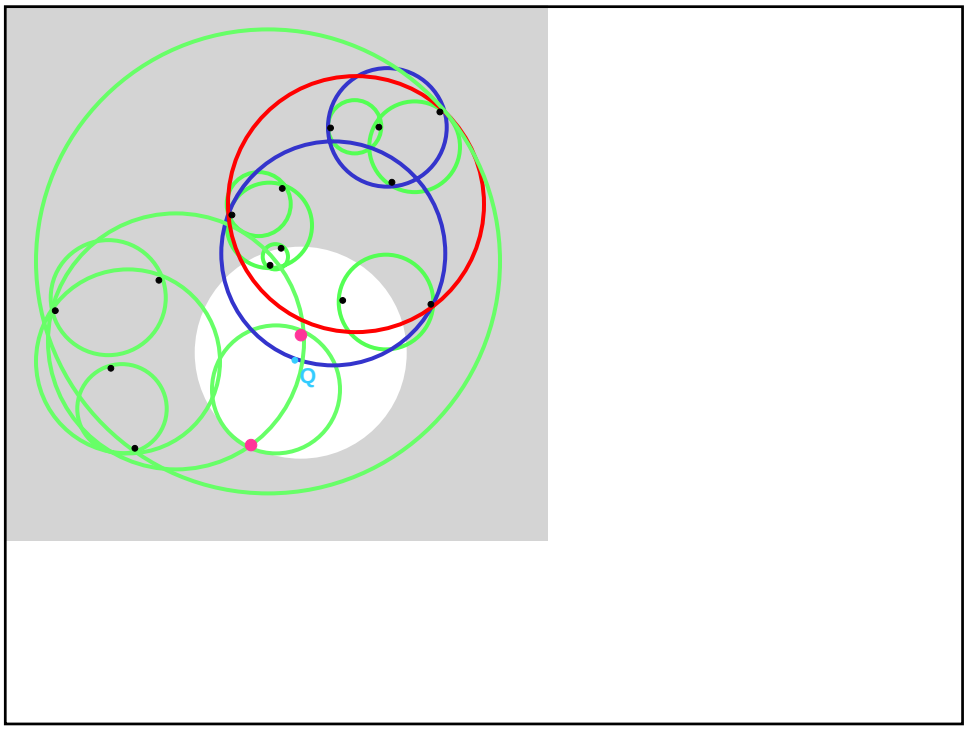

Q

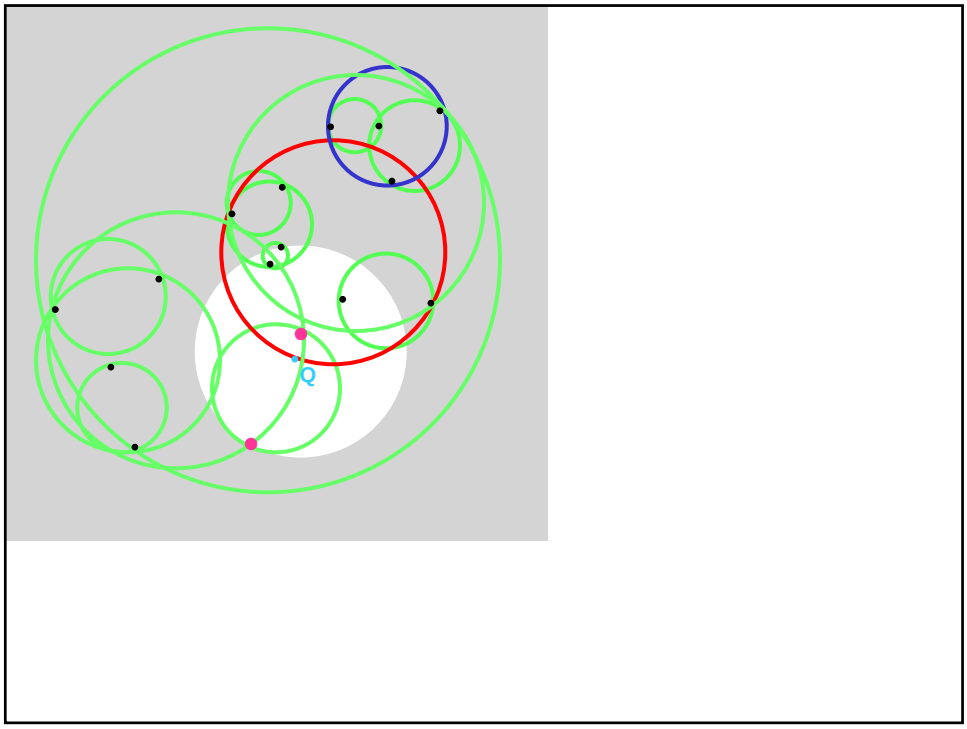

Q

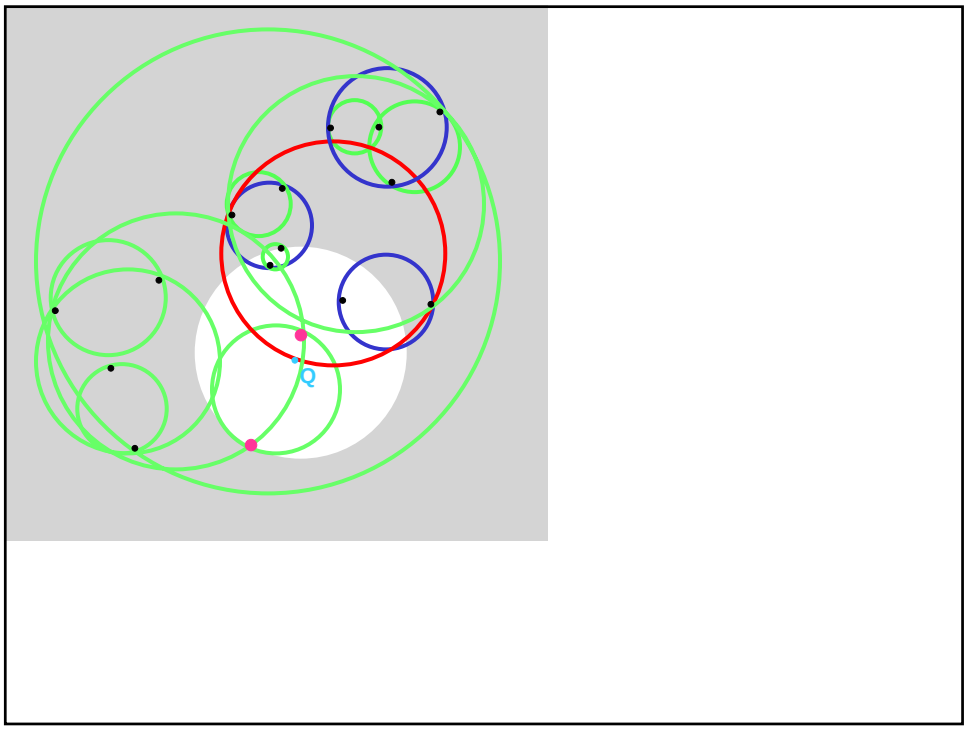

Q

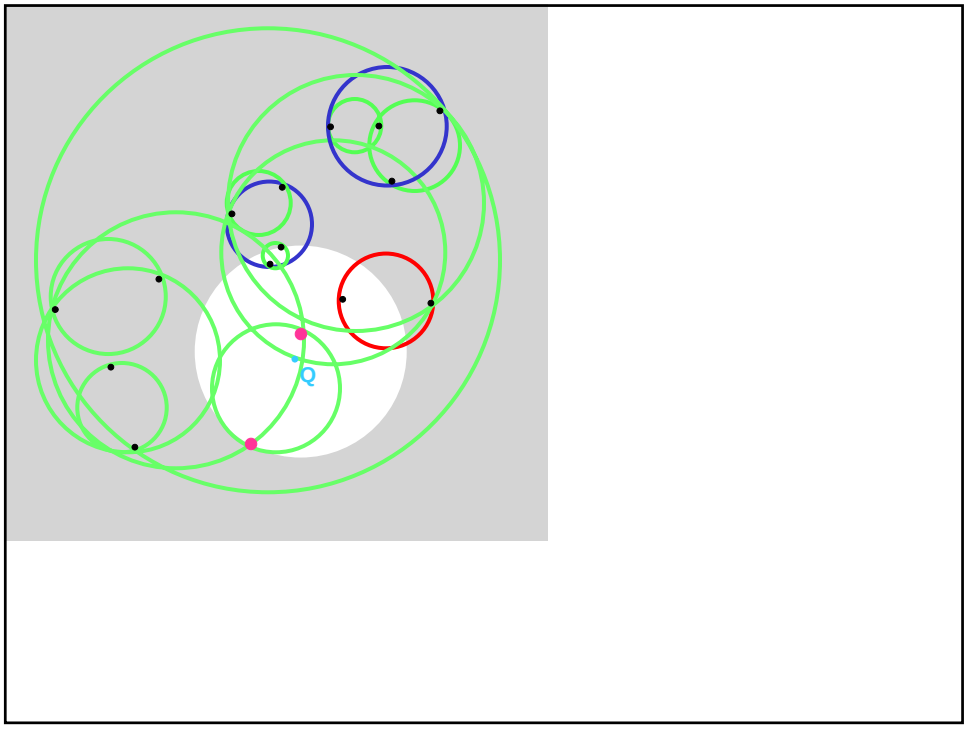

Q

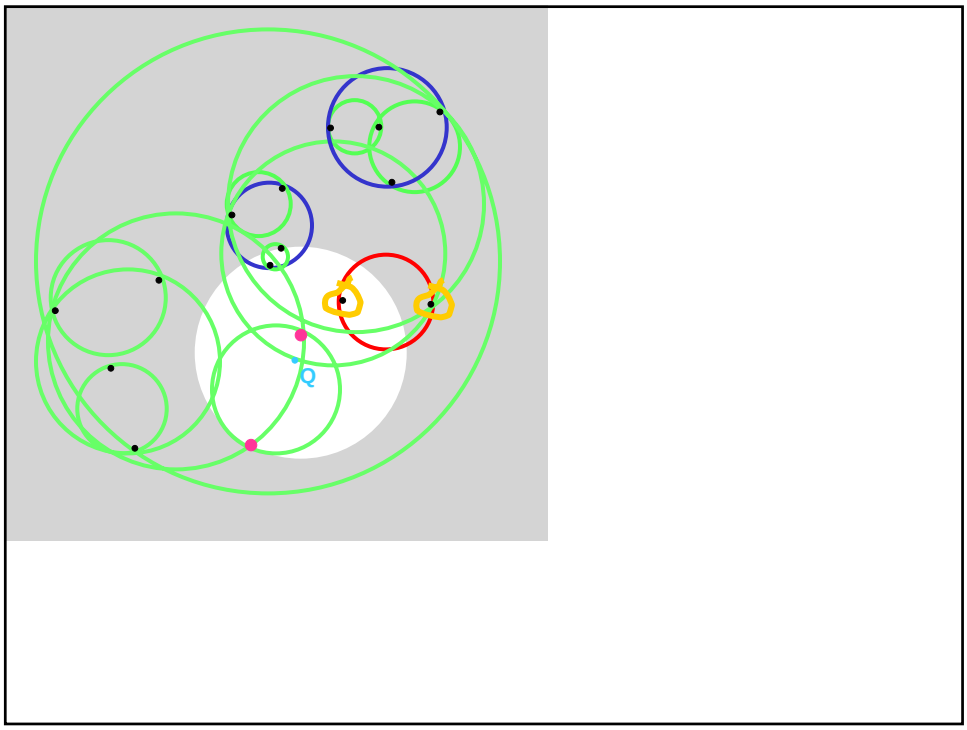

Q

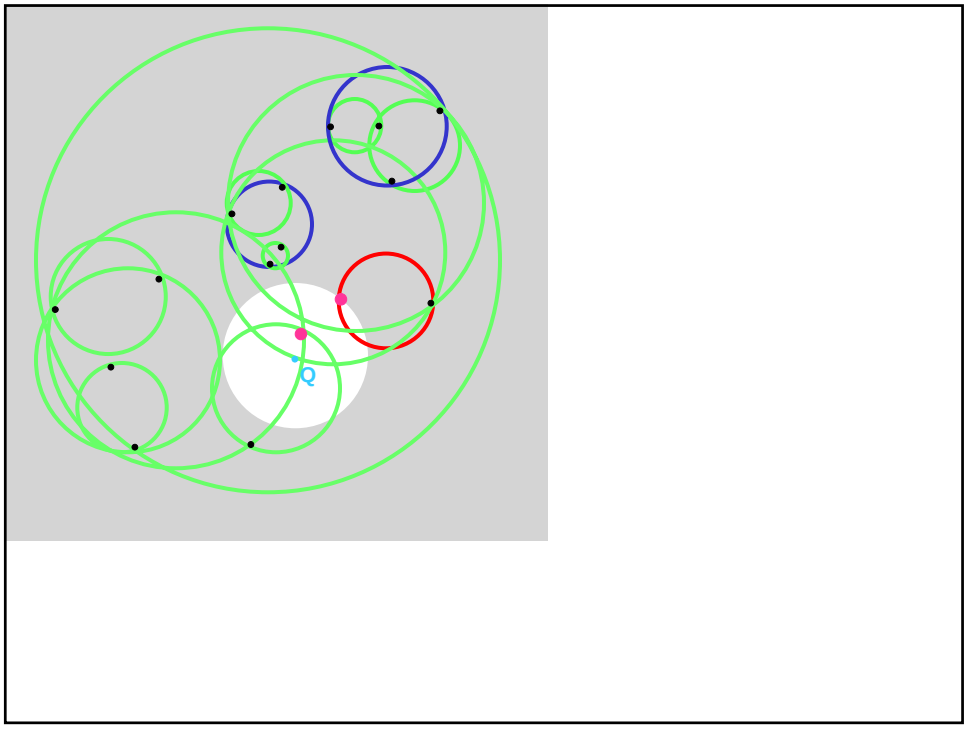

Q

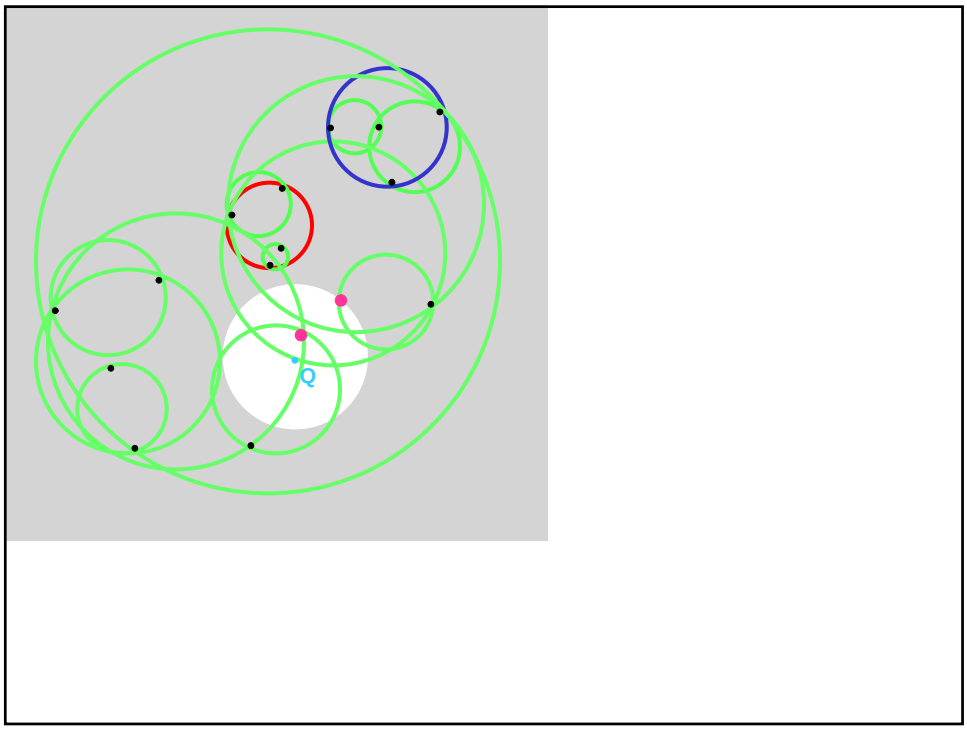

Q

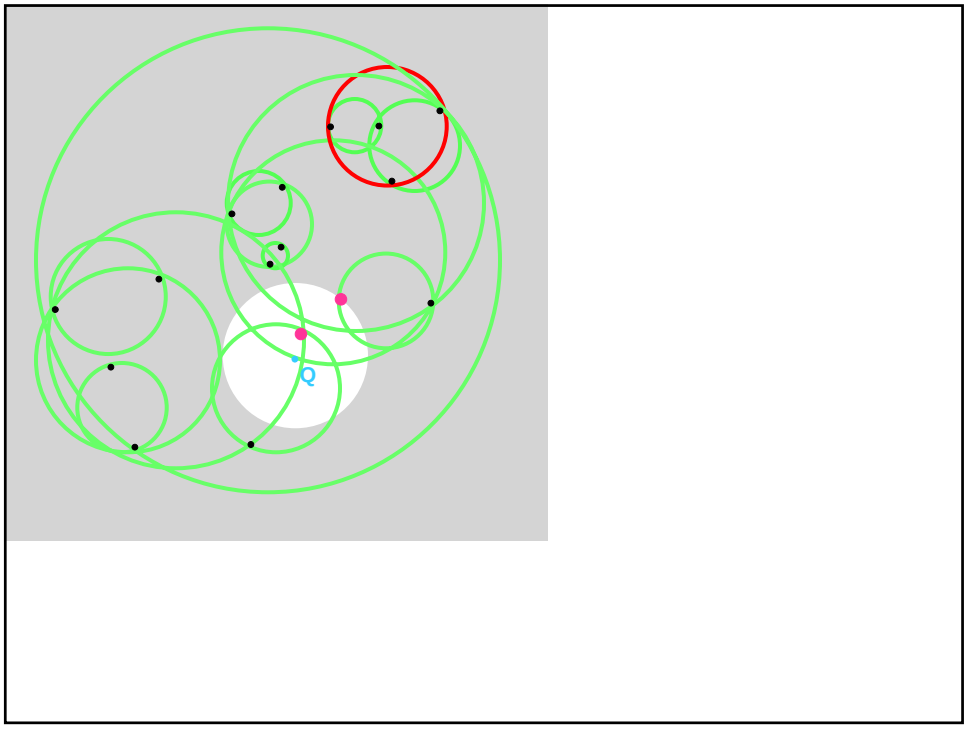

Q

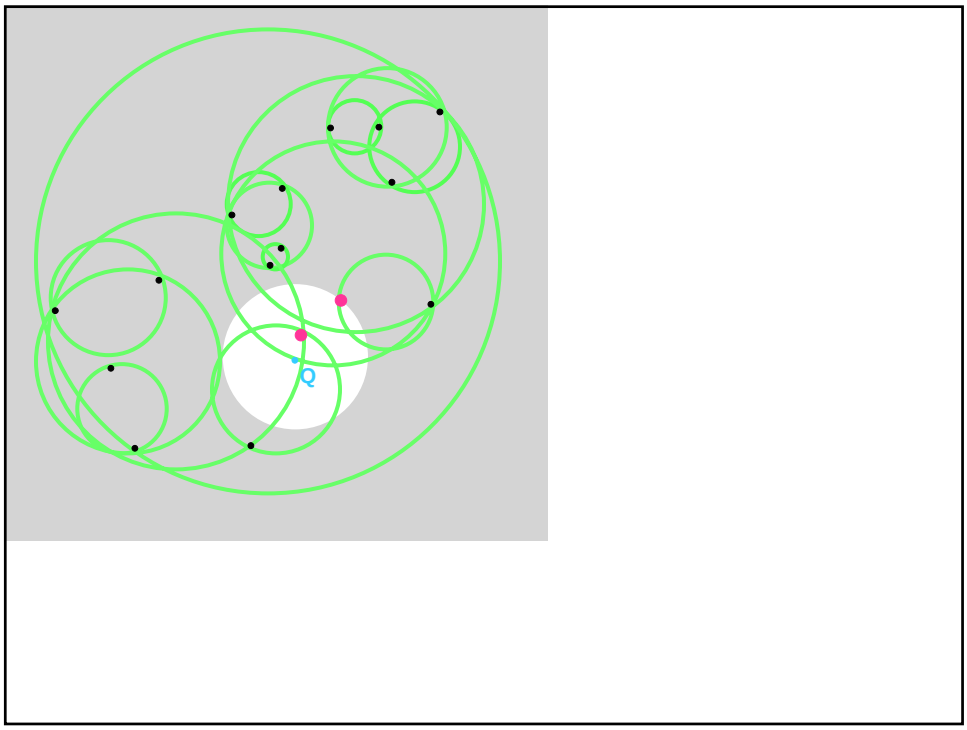

Q


Q


Q


Q


Q


Q

Outline

Cached Sufficient Statistics

Ball Trees Refresher

K-nearest-neighbor classification (exploiting the question part one)

Non-parametric classification

Biosurveillance and Epidemiology

Scan Statistics (exploiting the question part two)

Bayesian Network Learning

Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

Sensible conclusion

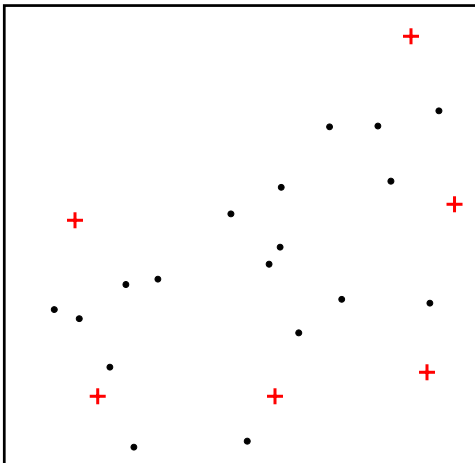
Flaky conclusion

KNS2

- Assume binary output
- Assume positive class is much less frequent than negative class
- Assume we want more than a "positive/negative" prediction: we want to know exactly how many of the K-NN are from the +ve class

KNS2 does this without finding the K-NN

collaboration with Ting Liu <tingliu@cs.cmu.edu>

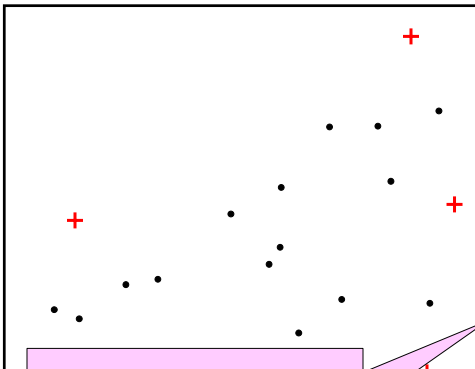


Assume we have a set of data points.

Some are +ve points (denoted +)

The large majority are –ve points (denoted ●)

collaboration with Ting Liu <tingliu@cs.cmu.edu>



Assume we have a set of data points.

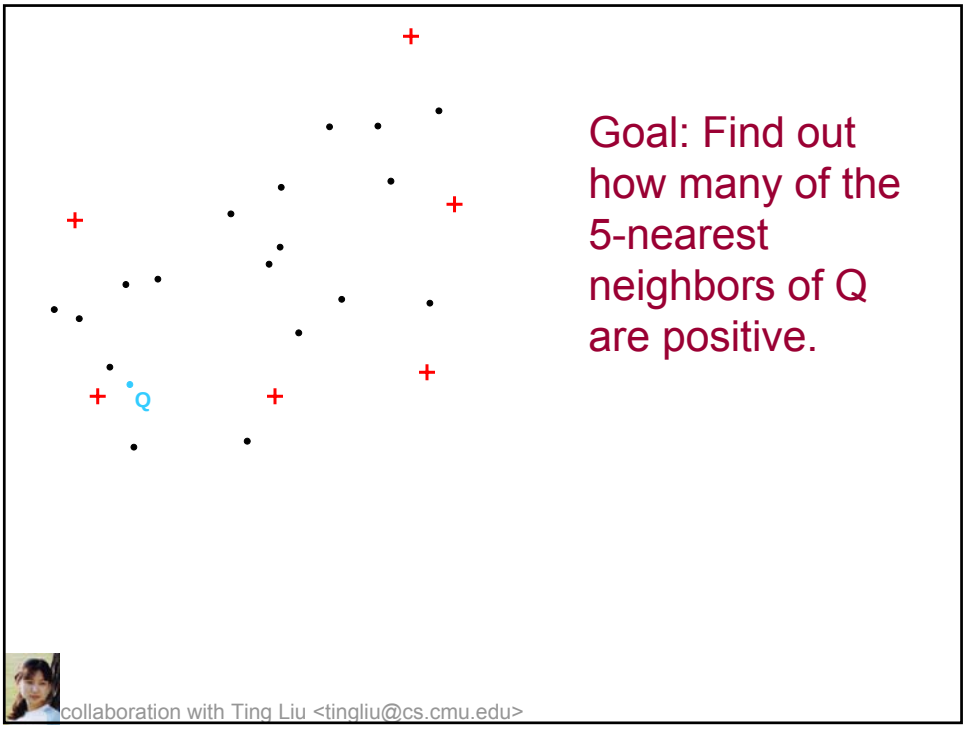
Some are +ve points (denoted +)

The vast majority are –ve points (denoted ●)

Preprocessing: Put all your +ve points in a small ball tree

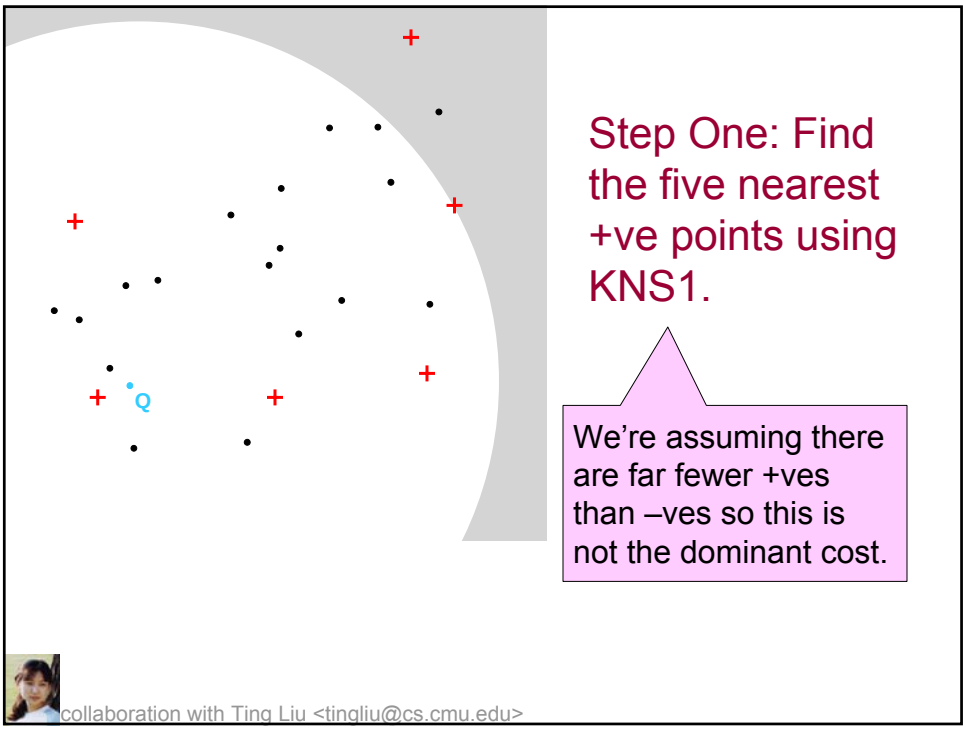
Preprocessing: Put all your -ve points in a separate large ball tree

collaboration with Ting Liu <tingliu@cs.cmu.edu>



Goal: Find out how many of the 5-nearest neighbors of Q are positive.

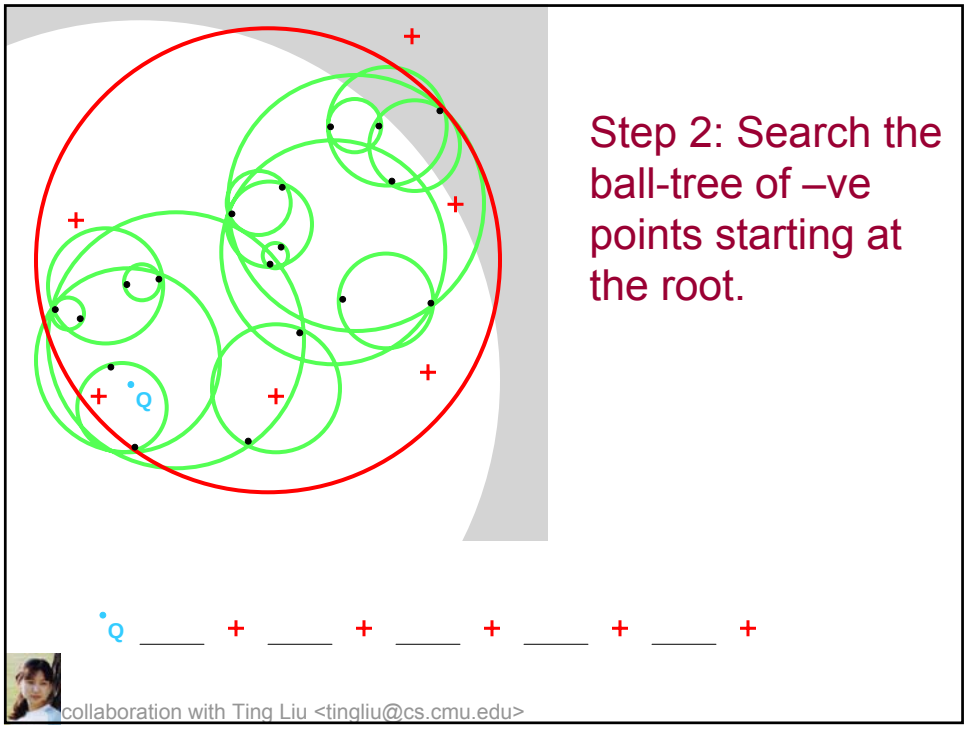
collaboration with Ting Liu <tingliu@cs.cmu.edu>



Step One: Find the five nearest +ve points using KNS1.

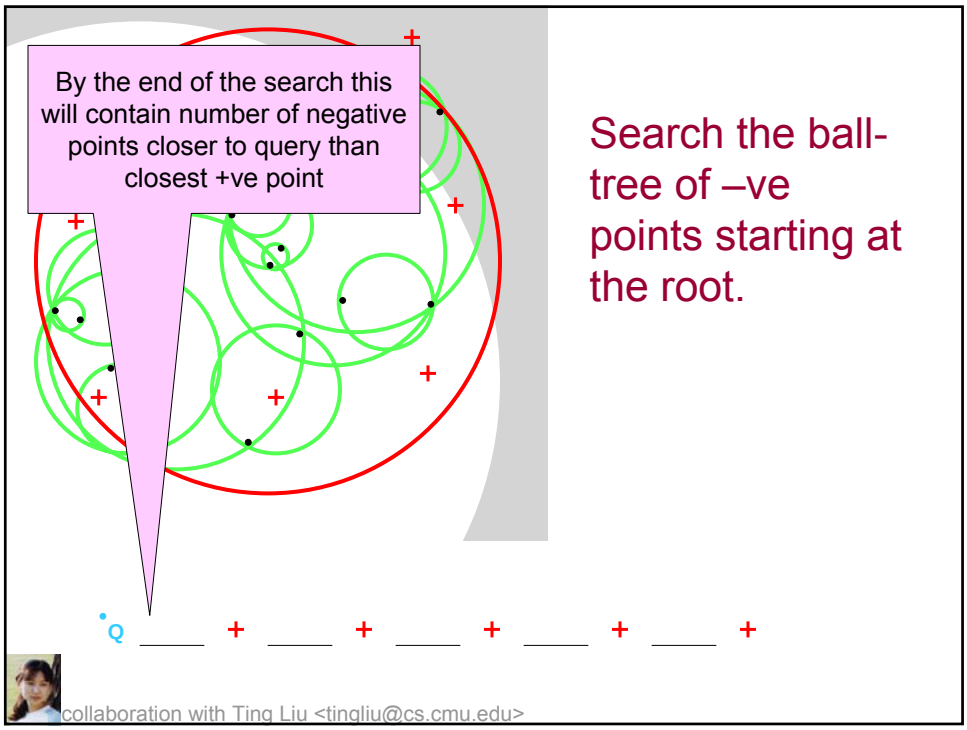
We're assuming there are far fewer +ves than –ves so this is not the dominant cost.

collaboration with Ting Liu <tingliu@cs.cmu.edu>



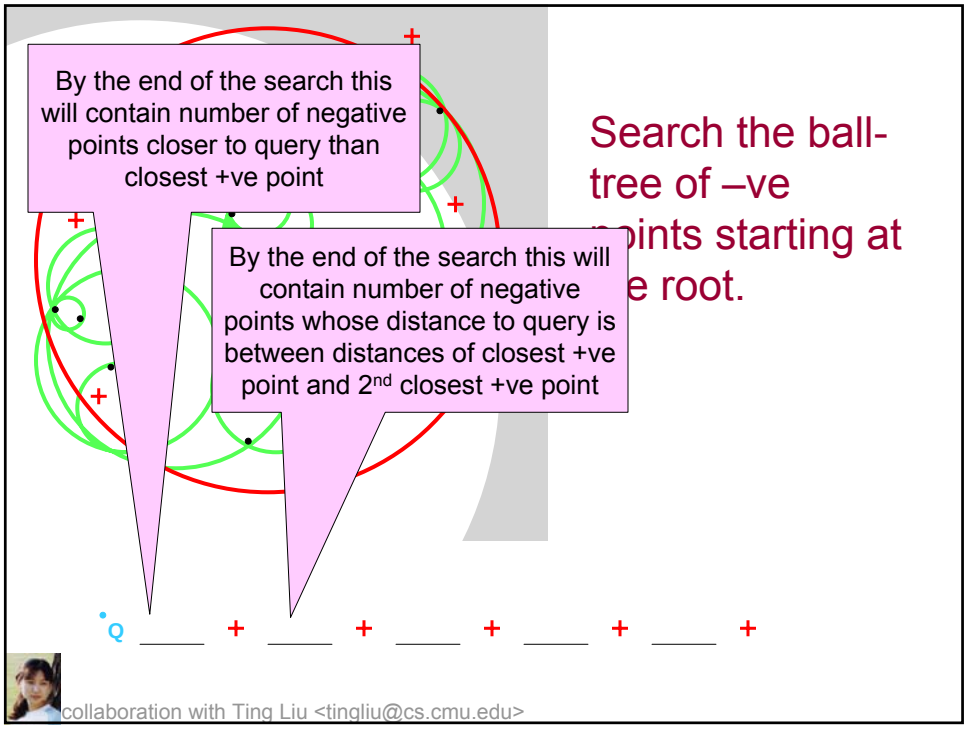
Step 2: Search the ball-tree of –ve points starting at the root.

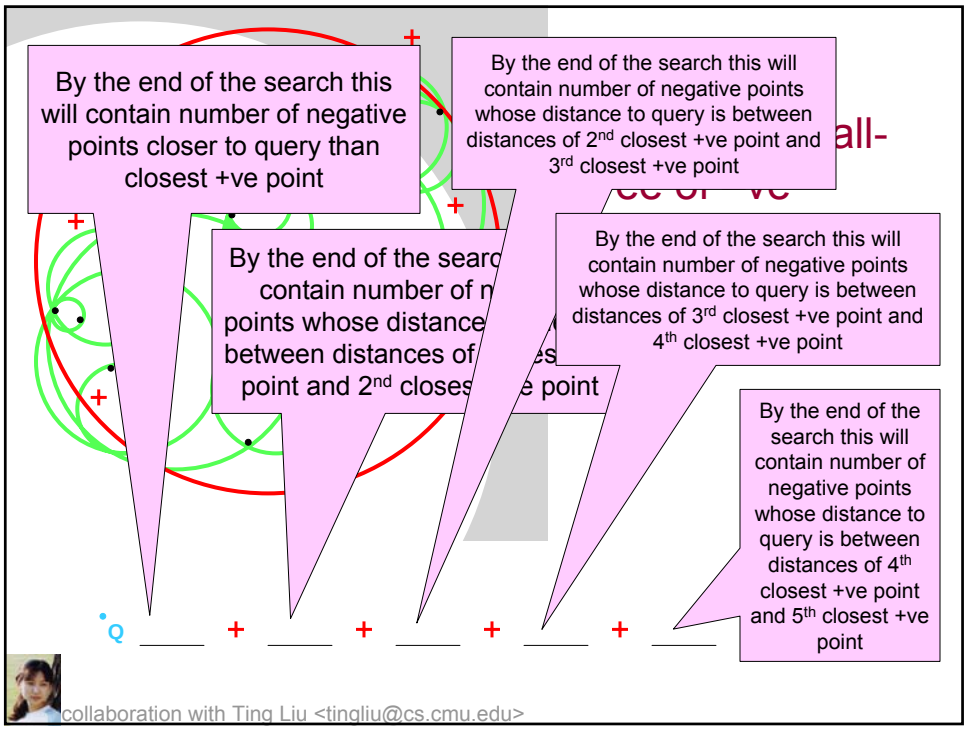
collaboration with Ting Liu <tingliu@cs.cmu.edu>

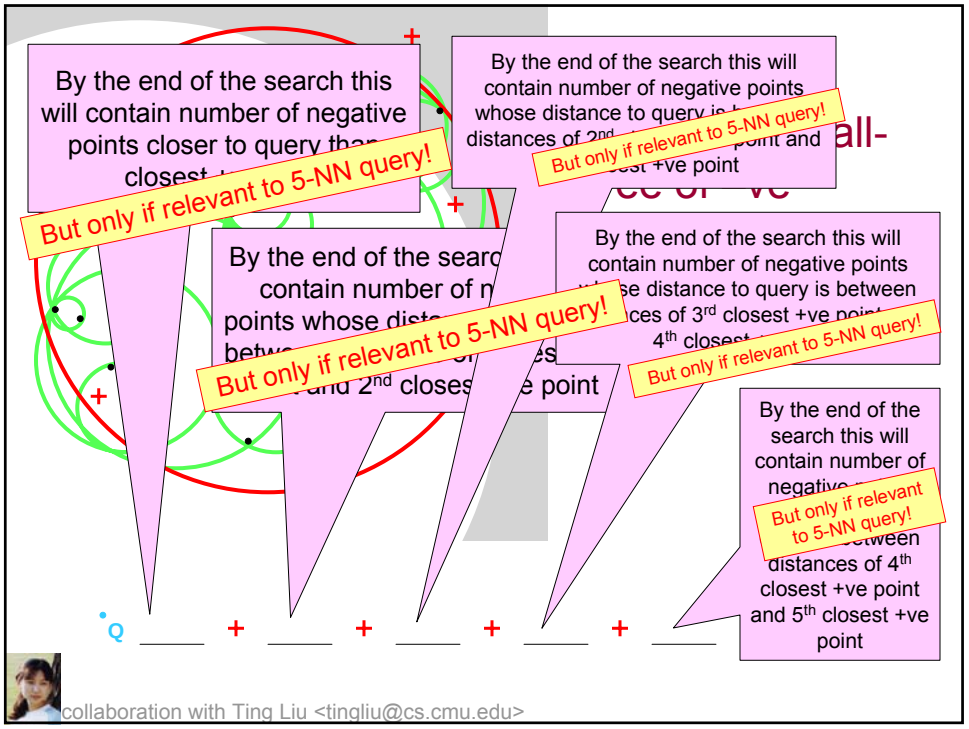


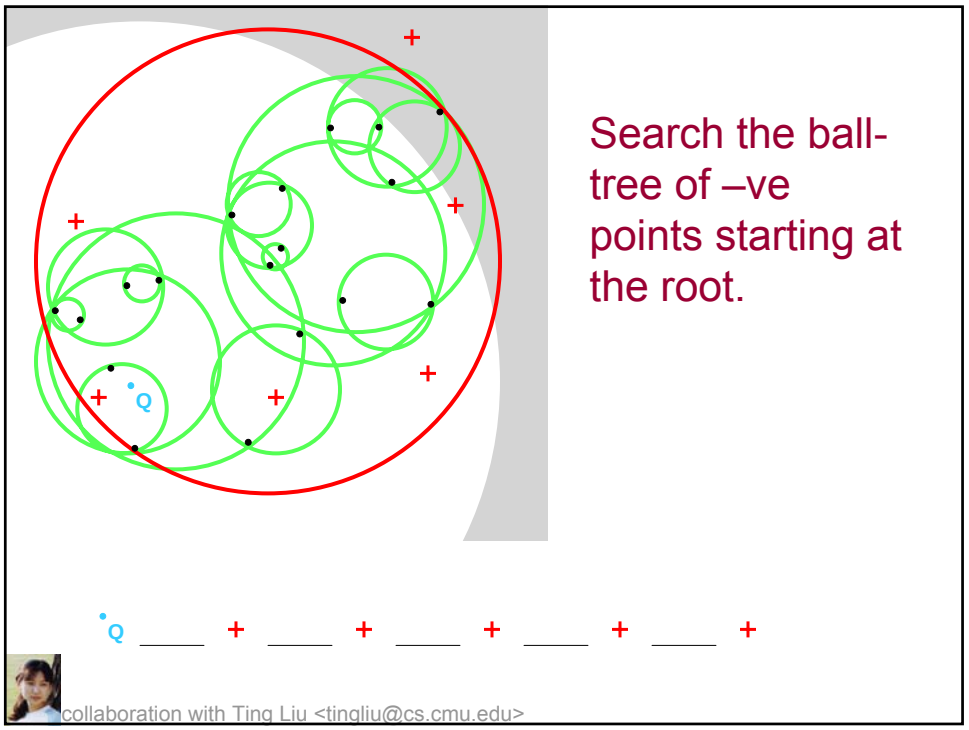
Search the ball-tree of –ve points starting at the root.

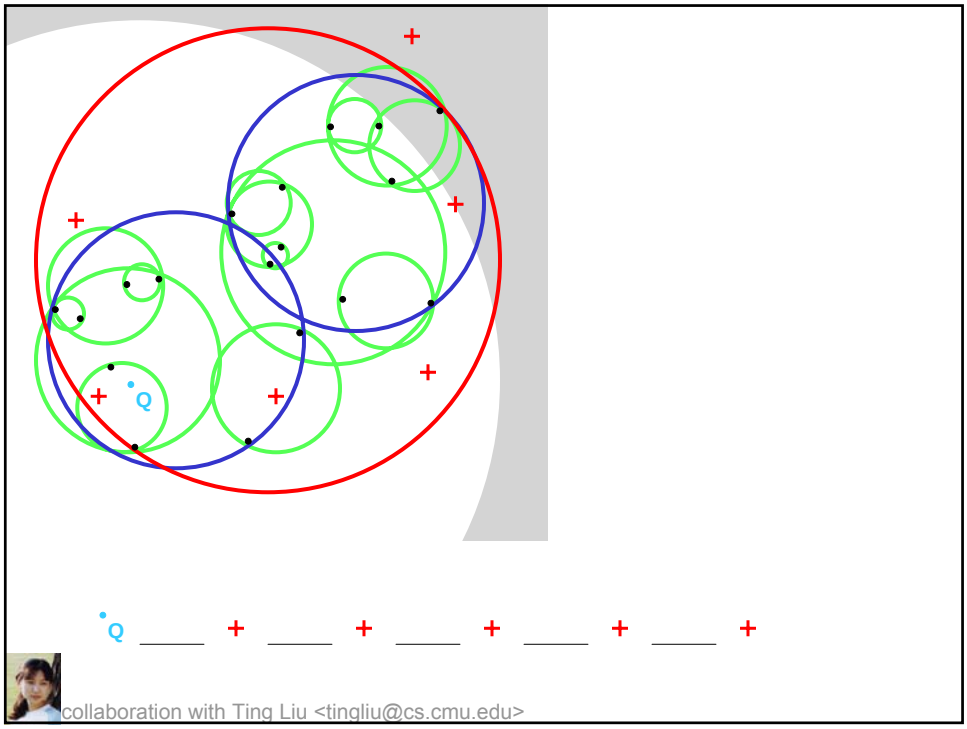
collaboration with Ting Liu <tingliu@cs.cmu.edu>

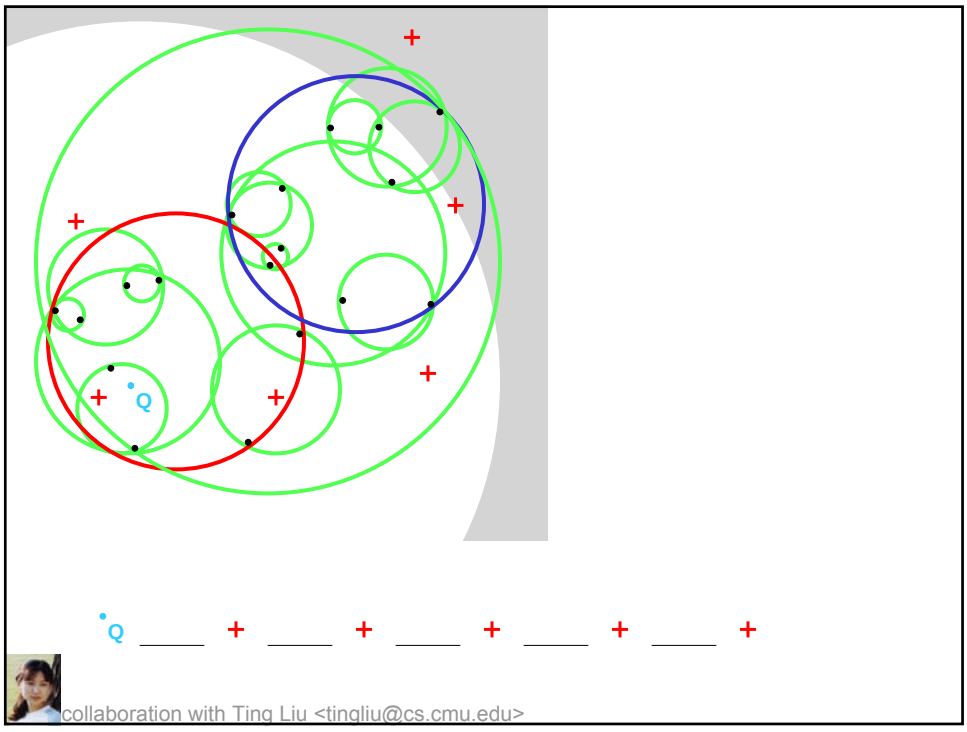

By the end of the search this will contain number of negative points closer to query than closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of closest +ve point and 2nd closest +ve point
Search the ball-tree of –ve
ints starting at
e root.
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

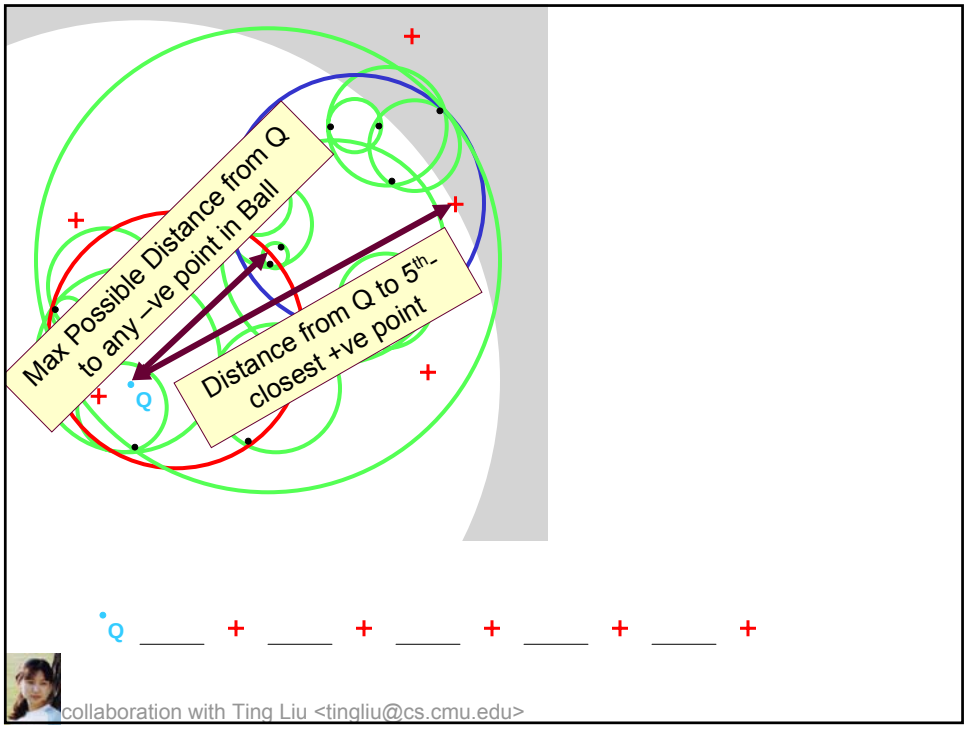

By the end of the search this will contain number of negative points closer to query than closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 2nd closest +ve point and 3rd closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 3rd closest +ve point and 4th closest +ve point
By the end of the search this will contain number of negative points whose distance to query is between distances of 4th closest +ve point and 5th closest +ve point
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>


By the end of the search this will contain number of negative points closer to query than closest
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between distances of 2nd
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between
But only if relevant to 5-NN query!
By the end of the search this will contain number of negative points whose distance to query is between distances of 4th closest +ve point and 5th closest +ve point
But only if relevant to 5-NN query!
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

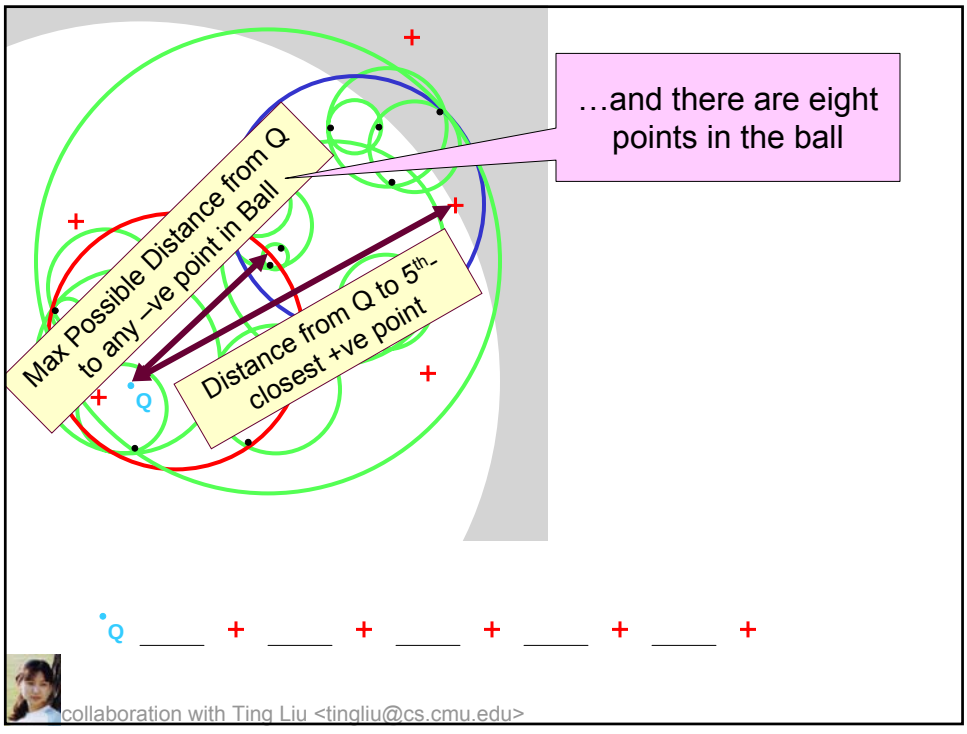

Search the ball-tree of –ve points starting at the root.
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>


Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

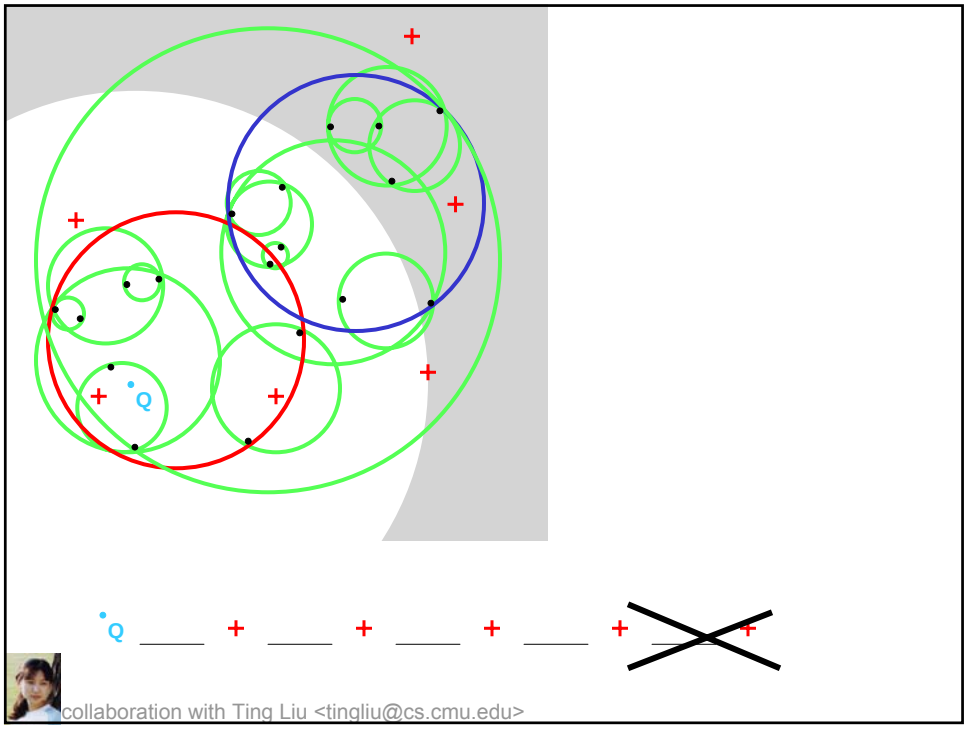

Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

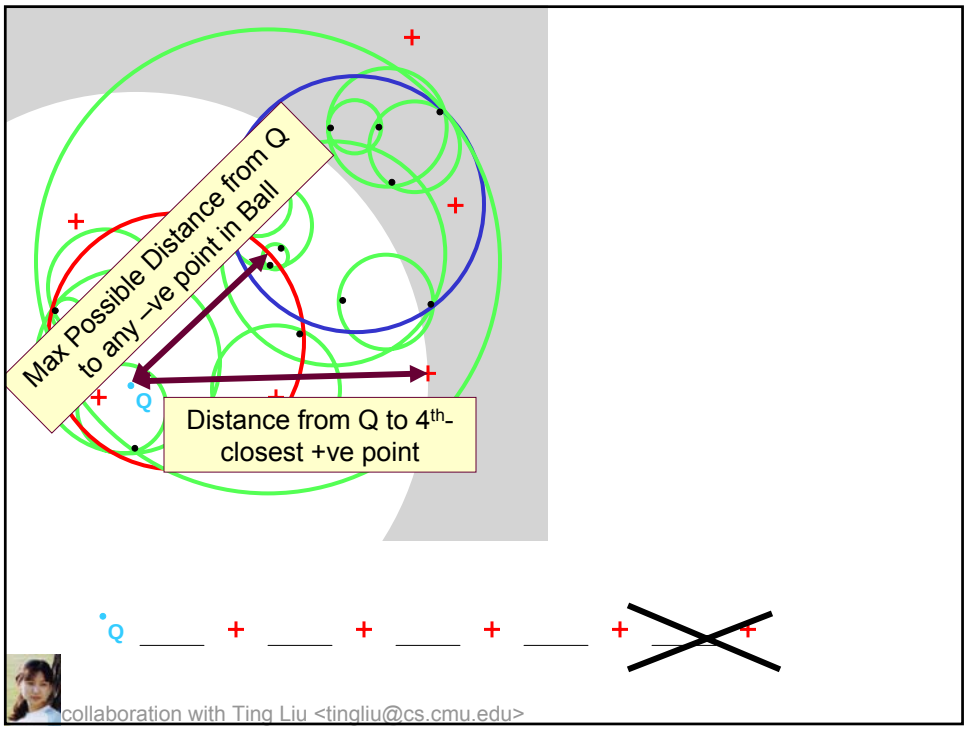


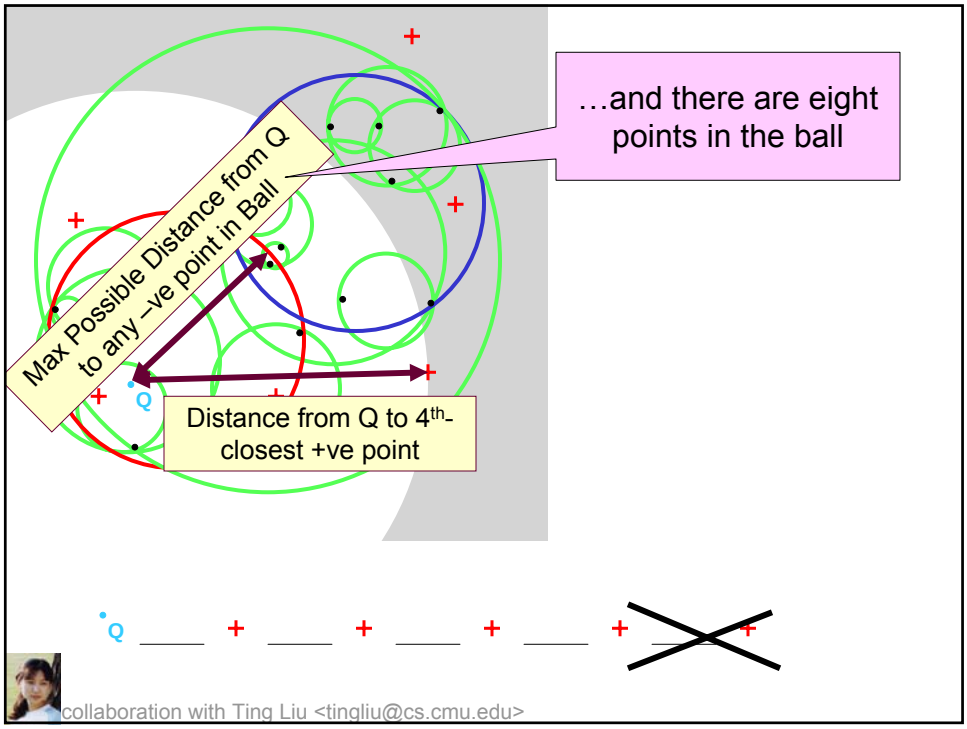
collaboration with Ting Liu <tingliu@cs.cmu.edu>



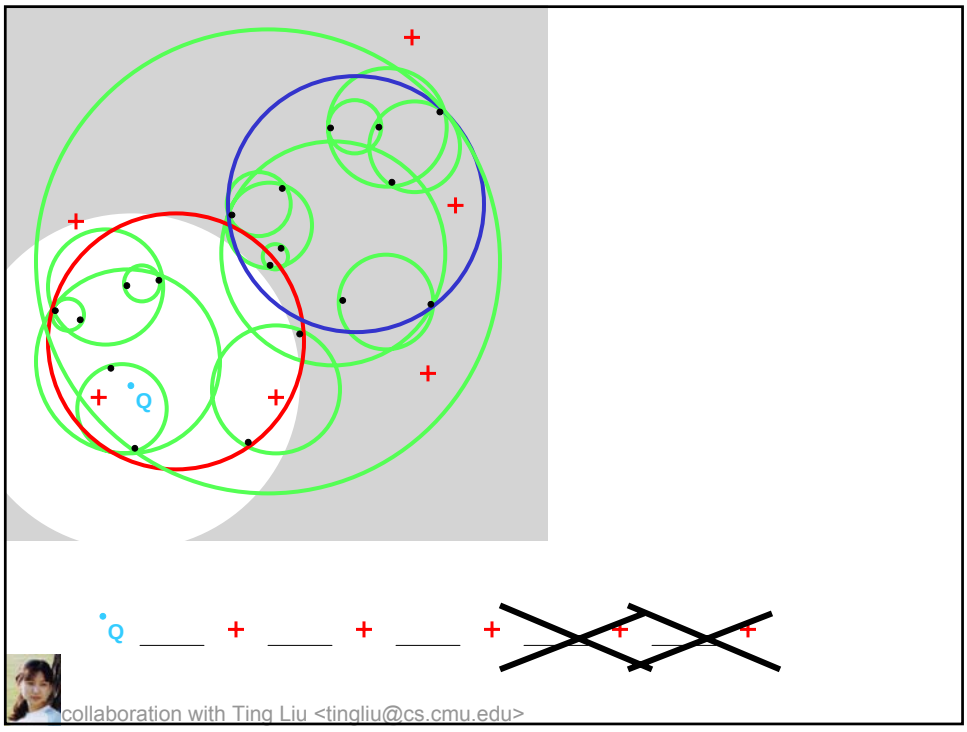
collaboration with Ting Liu <tingliu@cs.cmu.edu>


Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

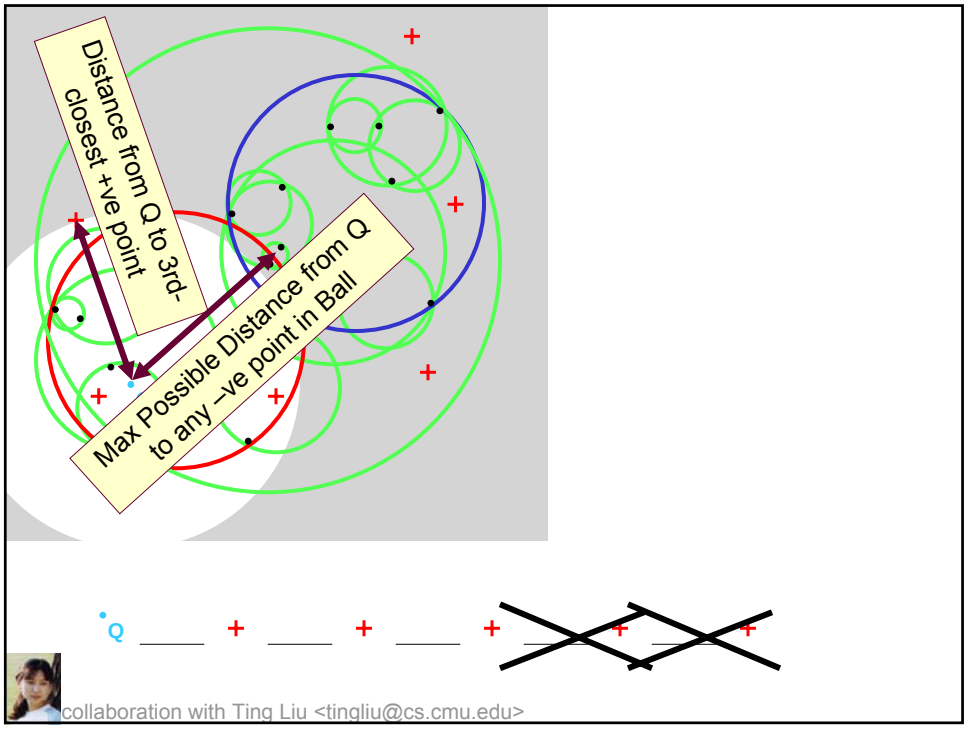

Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 4th-closest +ve point
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>



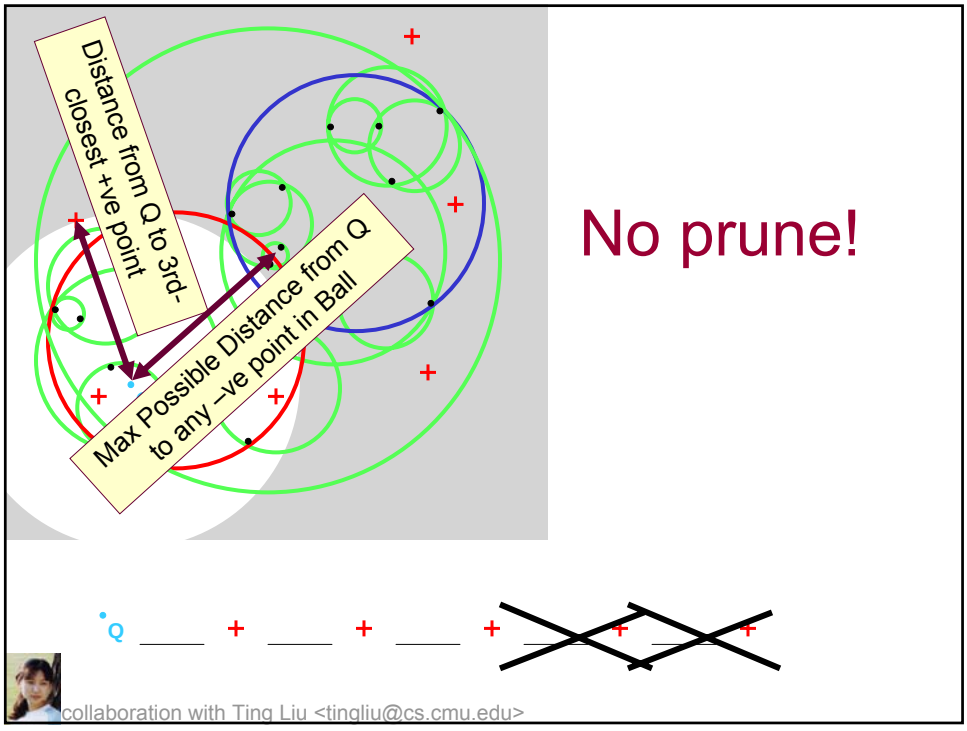
collaboration with Ting Liu <tingliu@cs.cmu.edu>



collaboration with Ting Liu <tingliu@cs.cmu.edu>

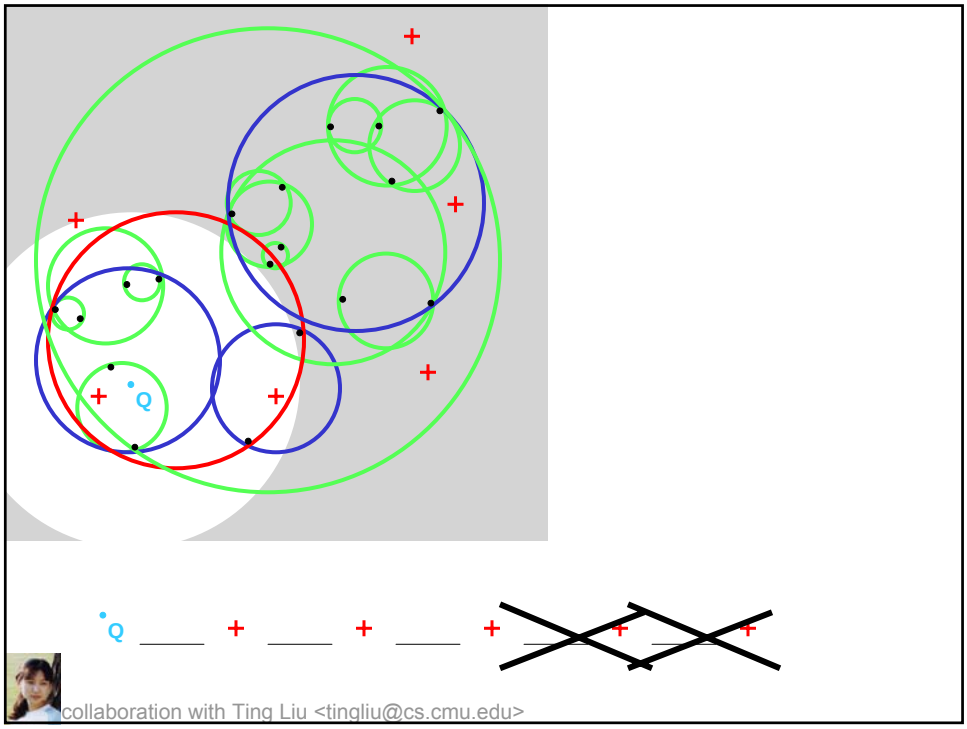


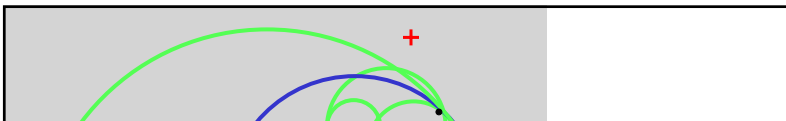
collaboration with Ting Liu <tingliu@cs.cmu.edu>



No prune!

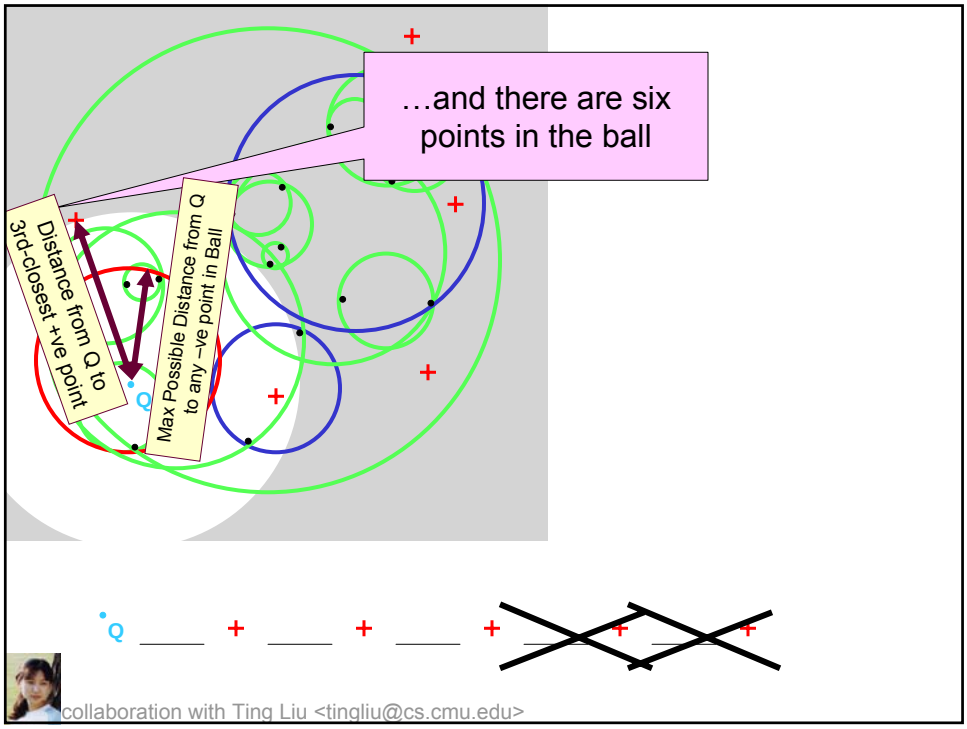
collaboration with Ting Liu <tingliu@cs.cmu.edu>



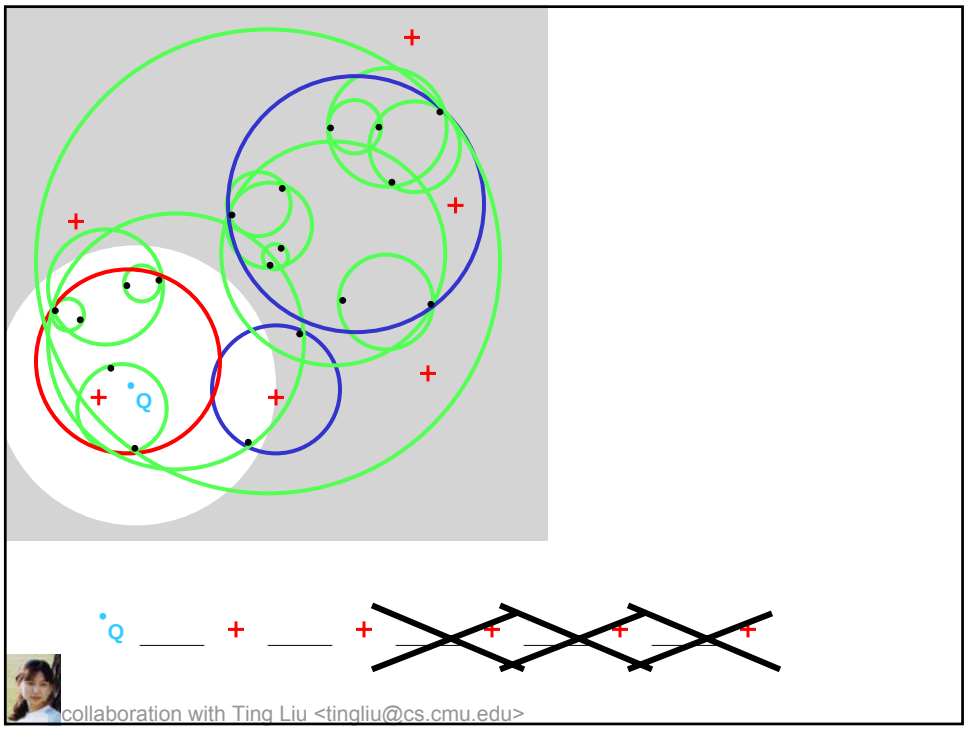


collaboration with Ting Liu <tingliu@cs.cmu.edu>

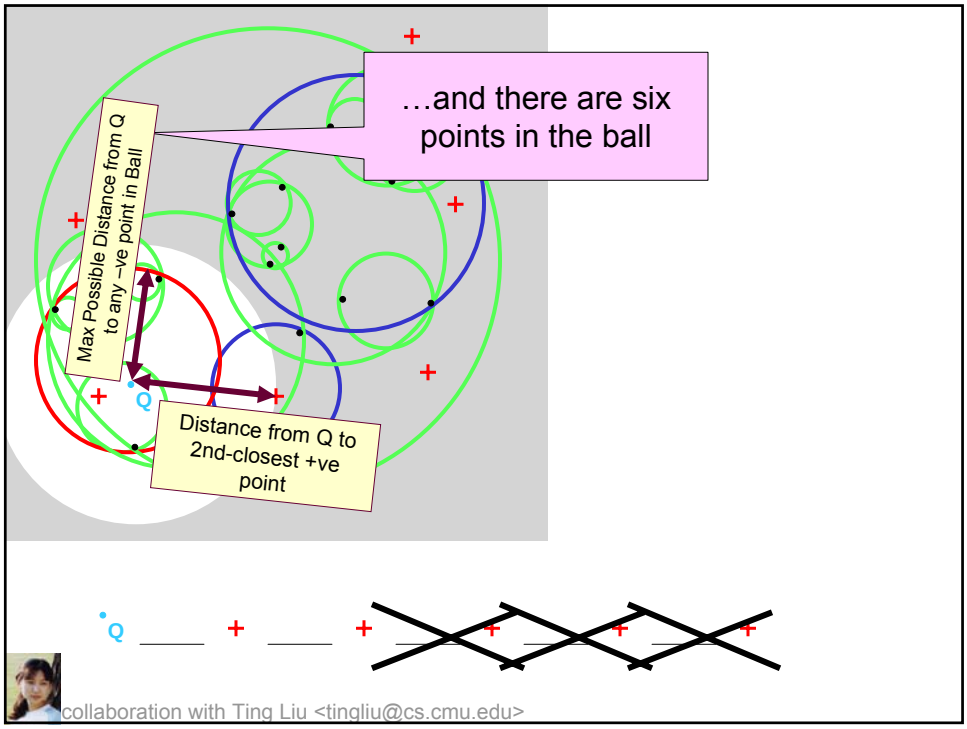
collaboration with Ting Liu <tingliu@cs.cmu.edu>

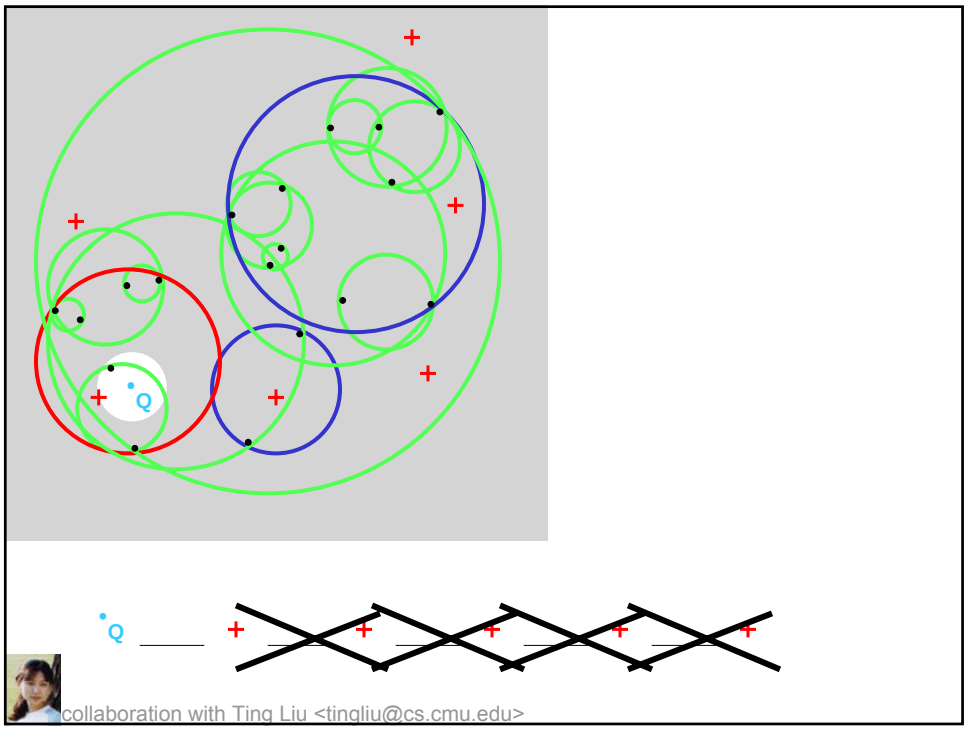


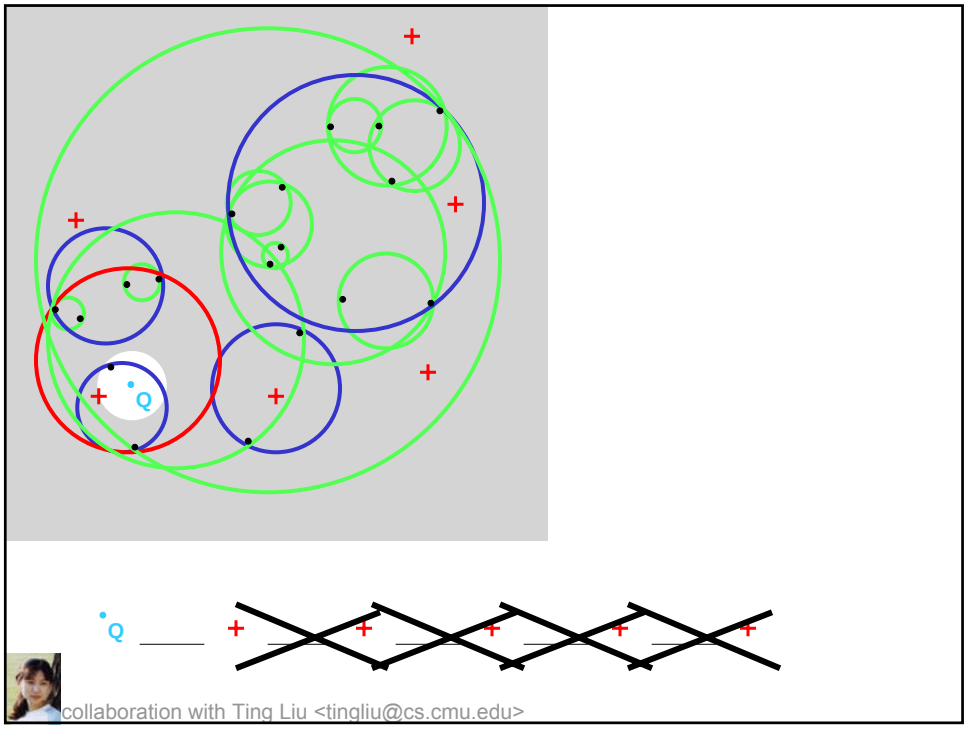
collaboration with Ting Liu <tingliu@cs.cmu.edu>



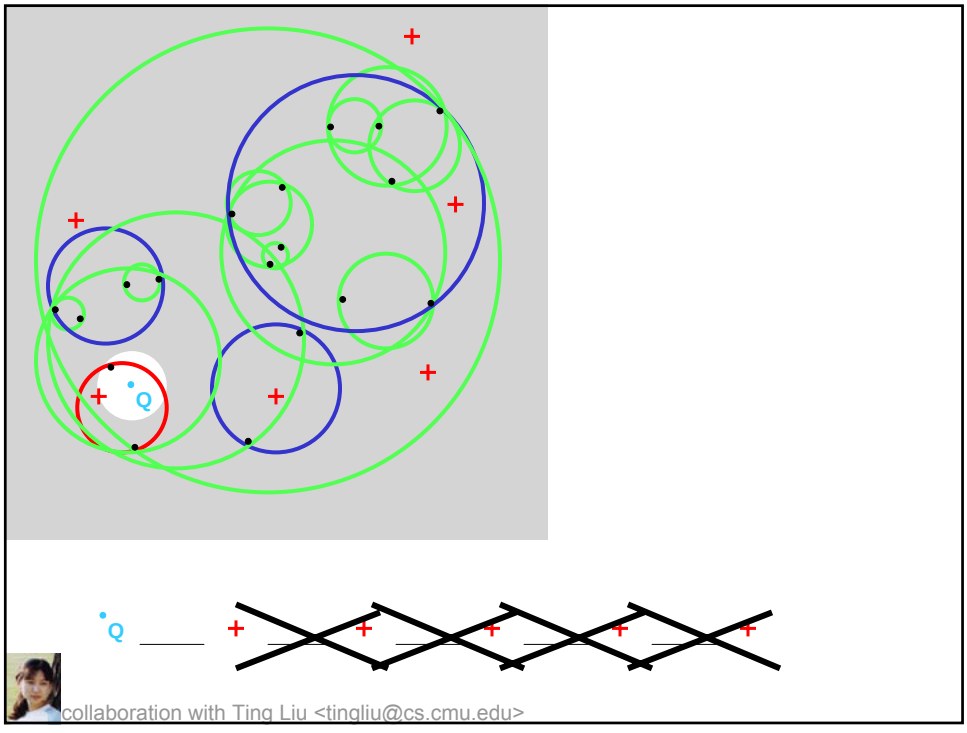
collaboration with Ting Liu <tingliu@cs.cmu.edu>


…and there are six points in the ball
Max Possible Distance from Q to any –ve point in Ball
Distance from Q to 2nd-closest +ve point
Q
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

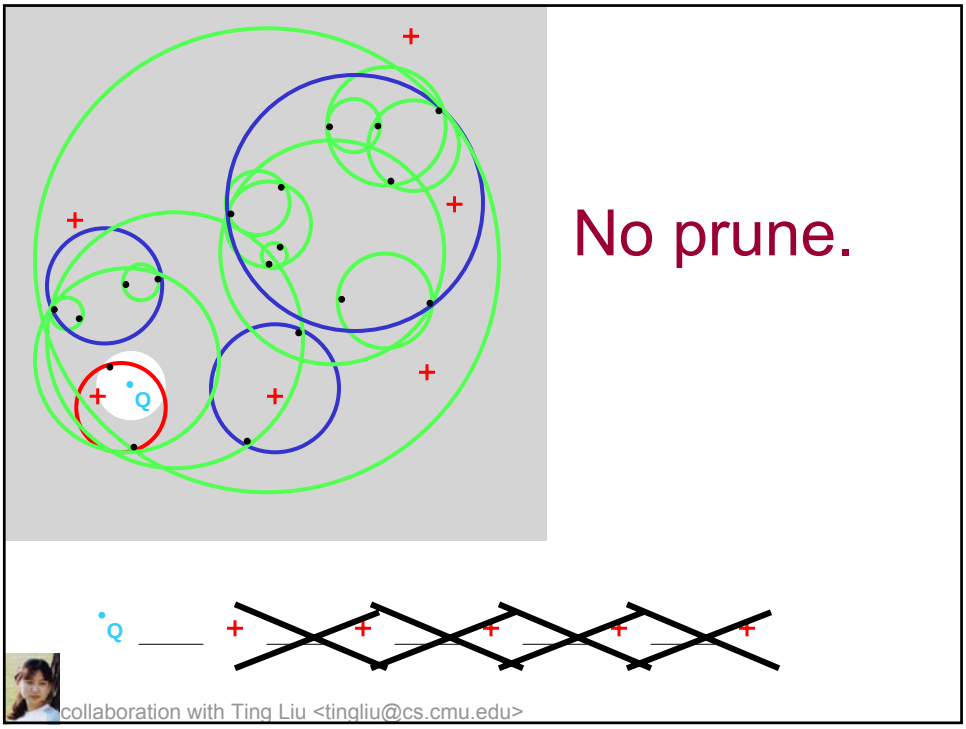

Q
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

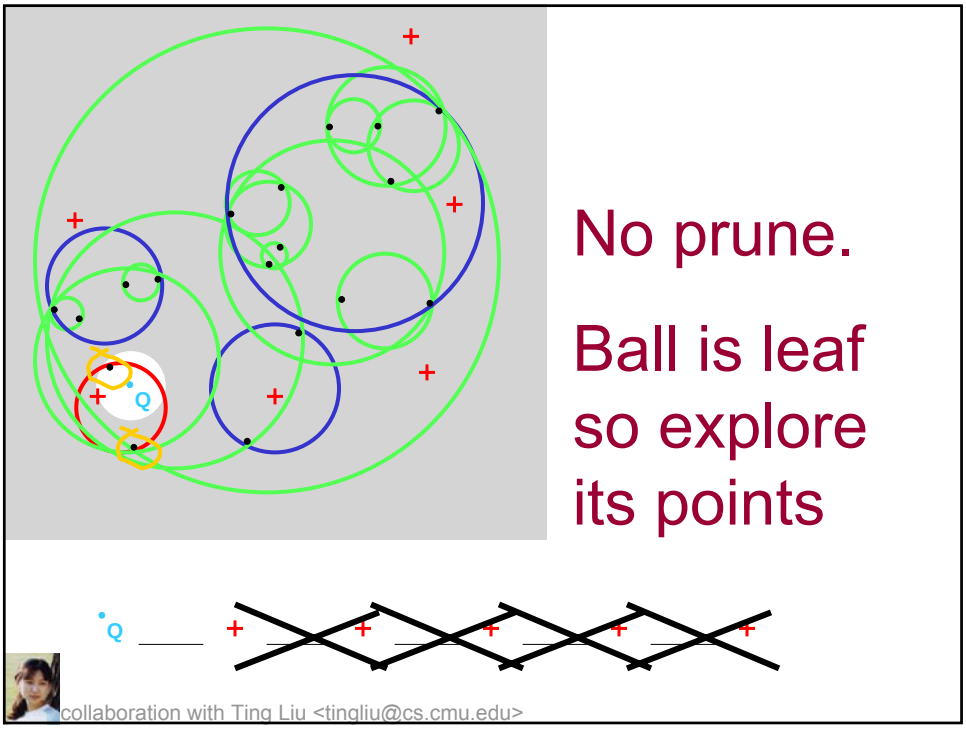


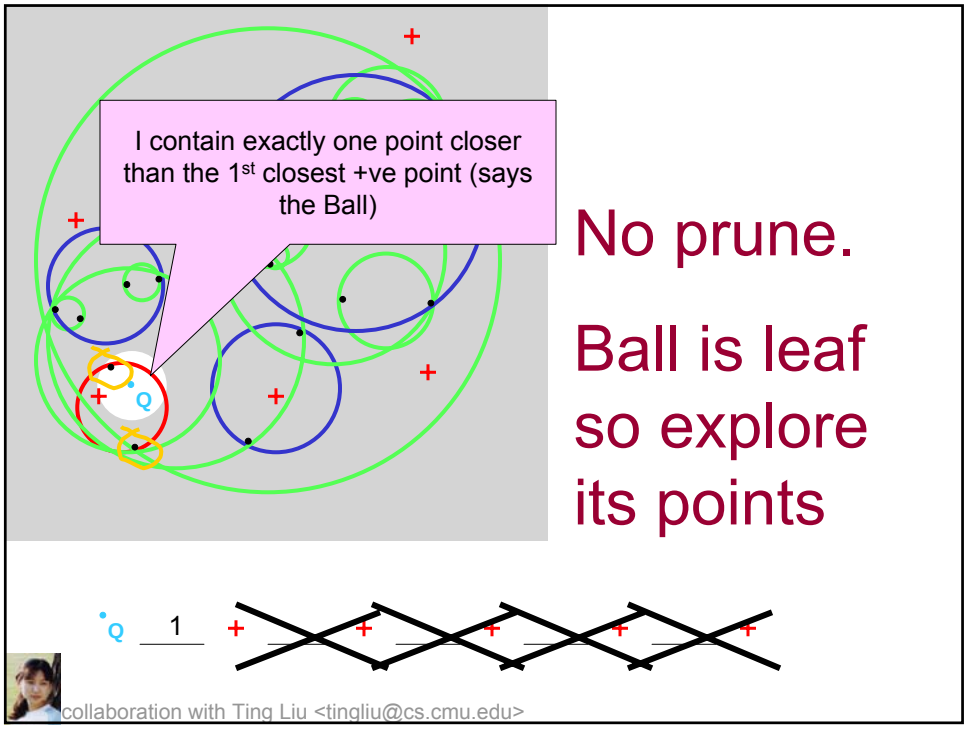
collaboration with Ting Liu <tingliu@cs.cmu.edu>



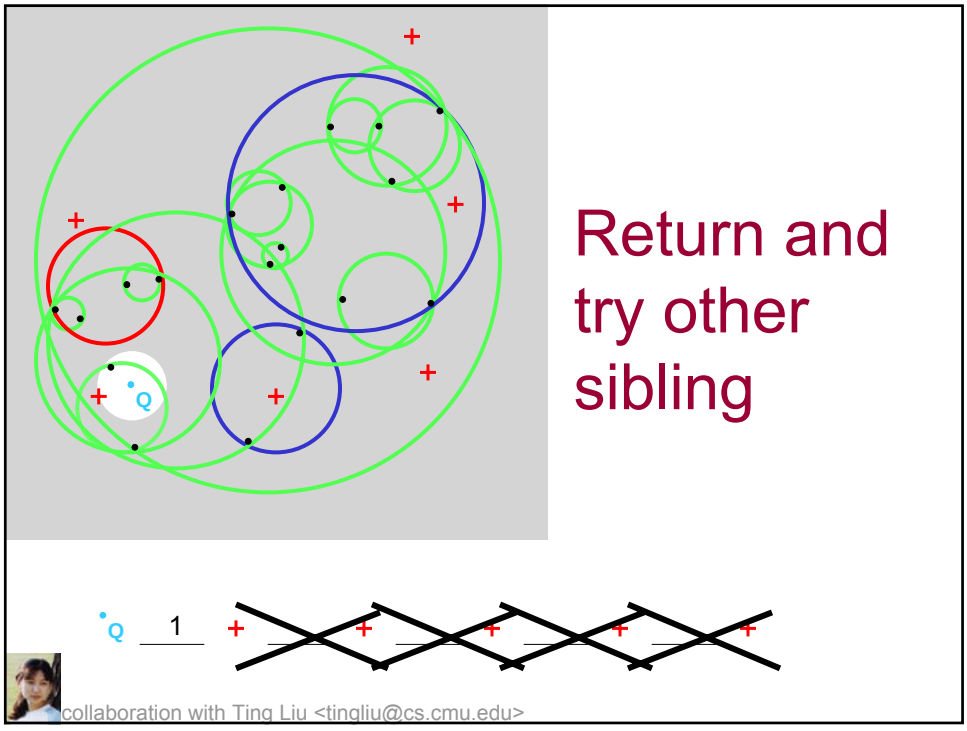
collaboration with Ting Liu <tingliu@cs.cmu.edu>


Q
No prune.
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>

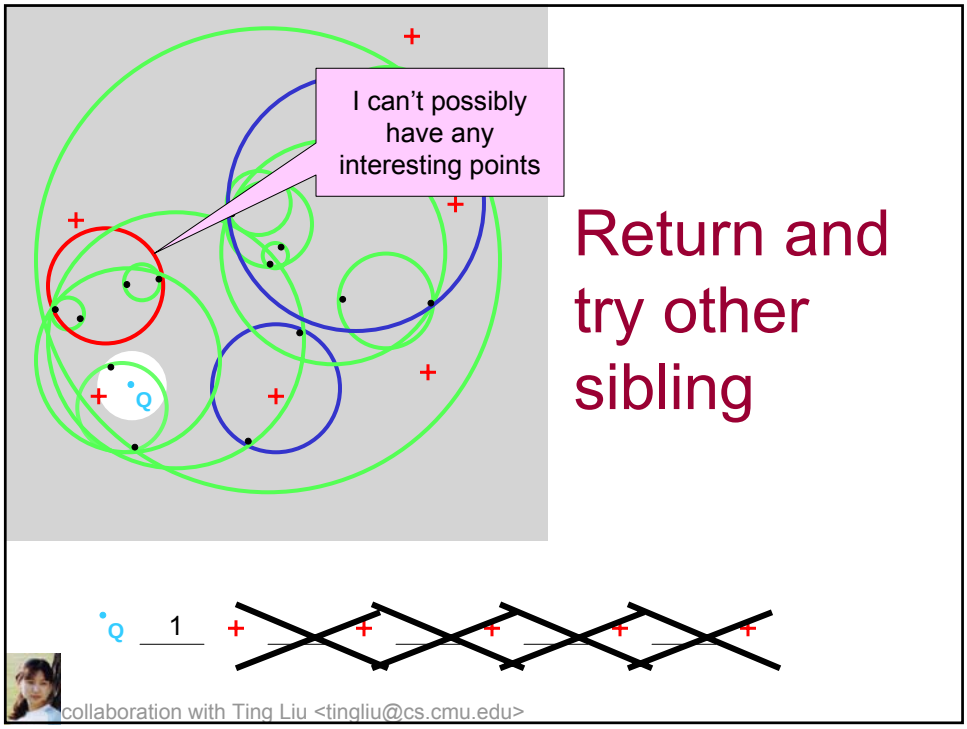

Q
No prune.
Ball is leaf so explore its points
Q
collaboration with Ting Liu <tingliu@cs.cmu.edu>



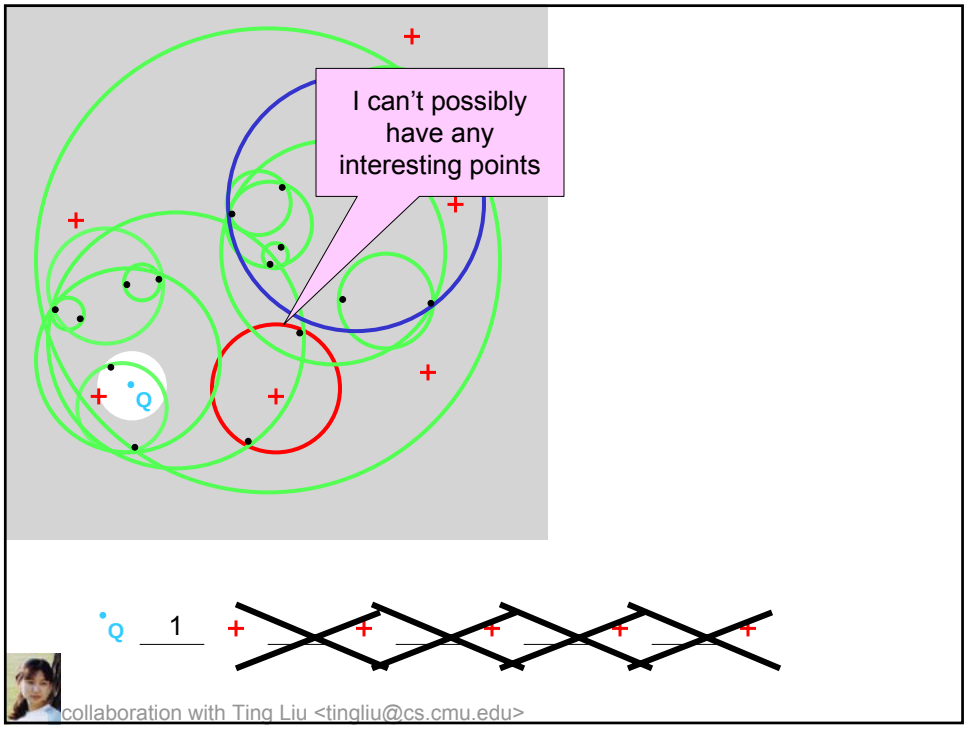
collaboration with Ting Liu <tingliu@cs.cmu.edu>

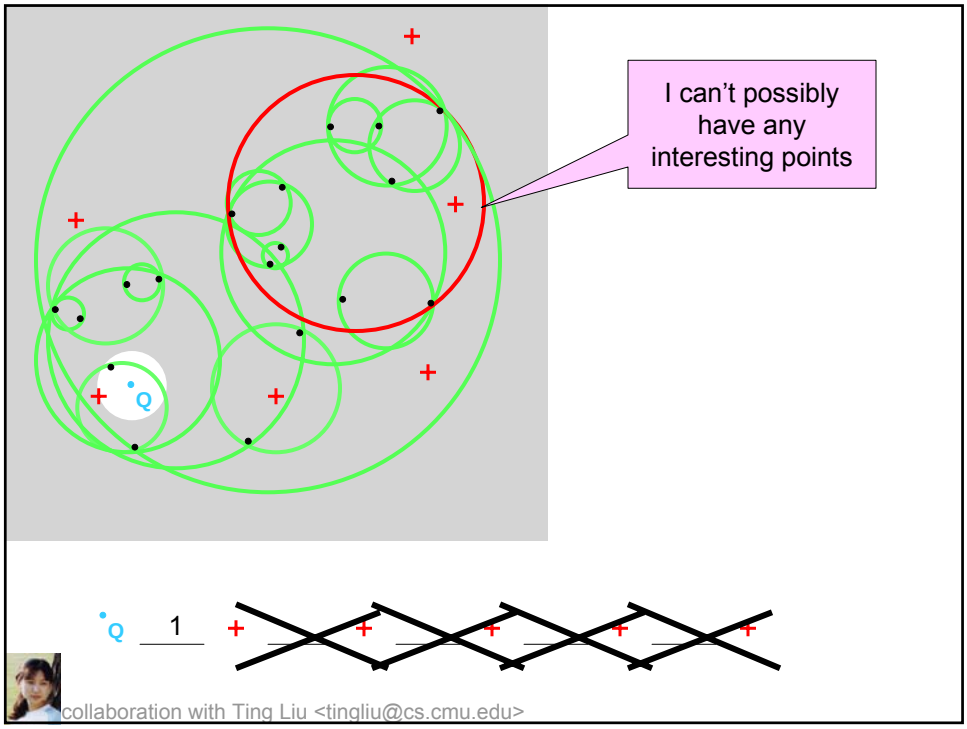


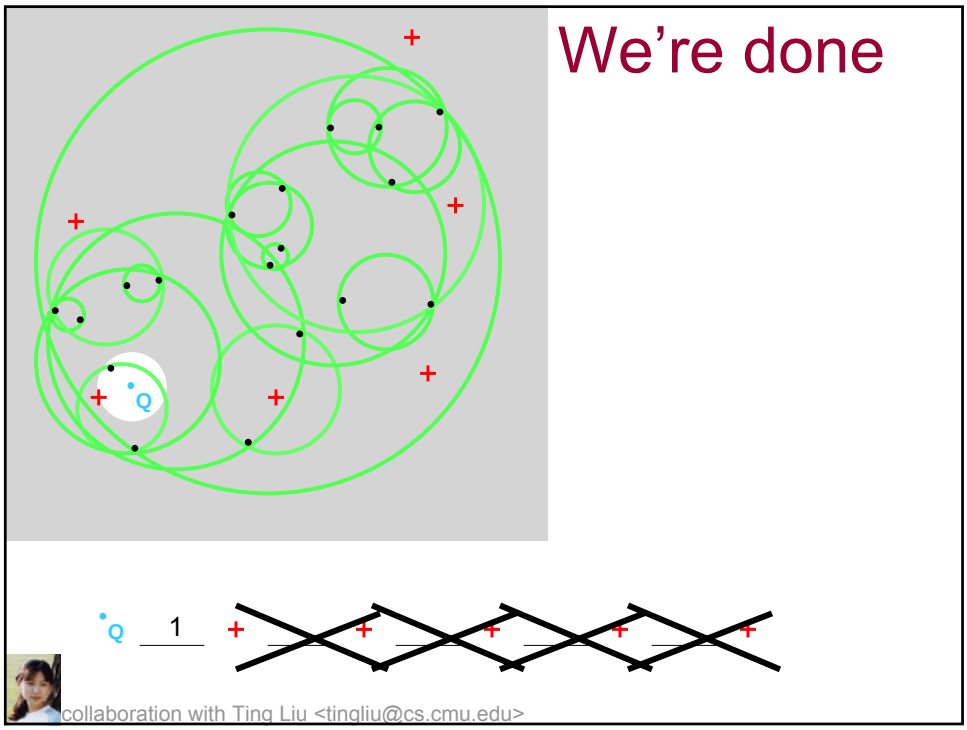
collaboration with Ting Liu <tingliu@cs.cmu.edu>

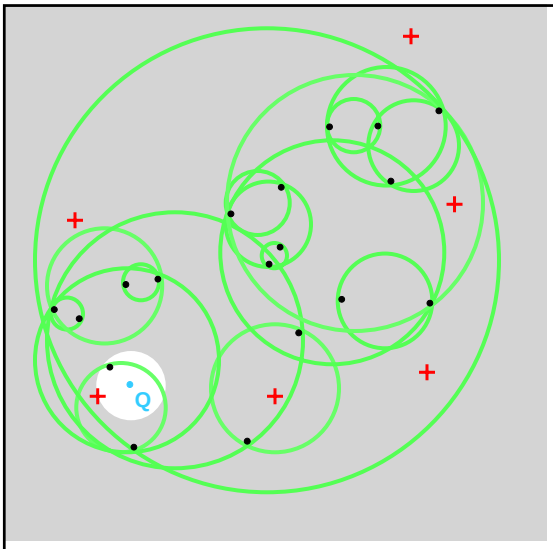


Return and try other sibling




I can't possibly have any interesting points
Q
1
collaboration with Ting Liu <tingliu@cs.cmu.edu>


We're done
Q
1
collaboration with Ting Liu <tingliu@cs.cmu.edu>



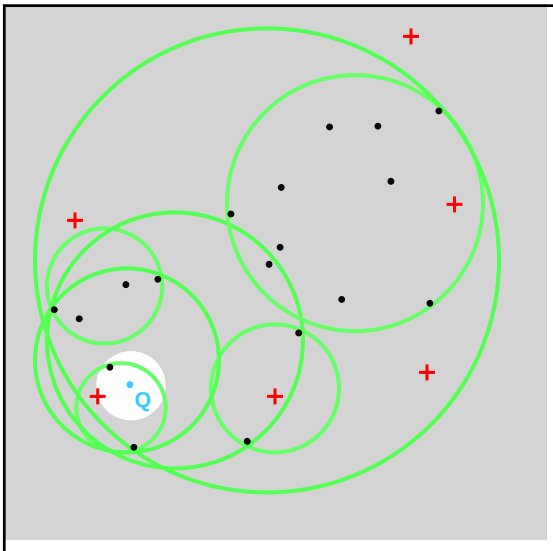
We're done

There's one –ve point closer than the closest +ve point.

There are more than 3 –ve points closer than the 2nd closest +ve point.

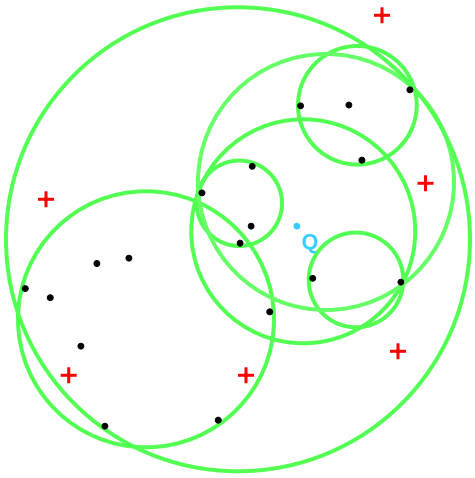
=> Exactly 1 of the 5 nearest neighbors is +ve

collaboration with Ting Liu <tingliu@cs.cmu.edu>



Balls visited

collaboration with Ting Liu <tingliu@cs.cmu.edu>



Another example

collaboration with Ting Liu <tingliu@cs.cmu.edu>

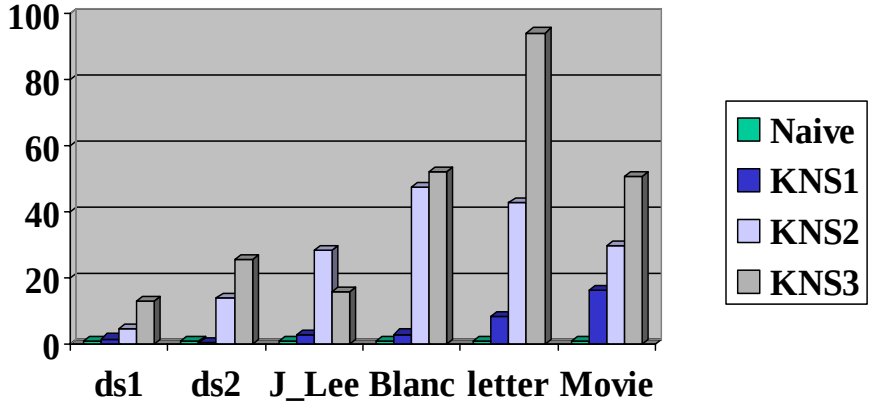
Experimental results

Dataset	Num. of records	Num. of Dimensions	Num.of positive	Num.pos/Num.neg
ds1	26733	6348	804	0.03
ds1.10pca	26733	10	804	0.03
ds1.100pca	26733	100	804	0.03
ds2	88358	1.1×10^6	211	0.002
ds2.100anchor	88358	100	211	0.002
J_Lee.100pca	181395	100	299	0.0017
Blanc_Mel	186414	10	824	0.004

Dataset	Num. records	Num. of Dimensions	Num.of positive	Num.pos/Num.neg
Letter	20000	16	790	0.04
Ipums	70187	60	119	0.0017
Movie	38943	62	7620	0.24
Kdd99(10%)	494021	176	97278	0.24

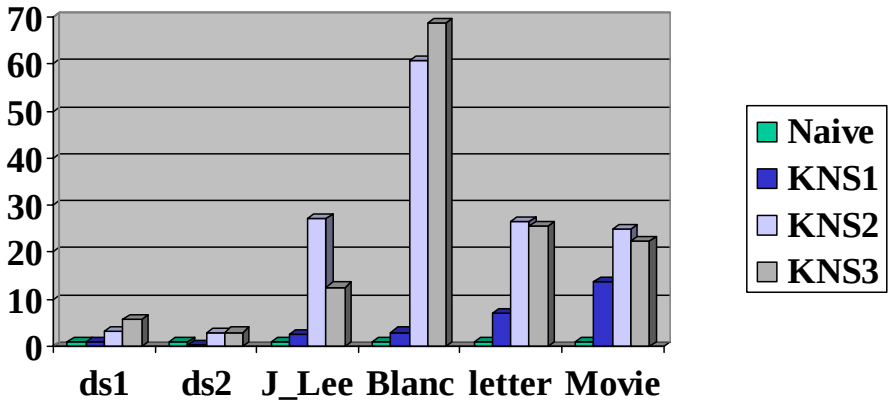
collaboration with Ting Liu <tingliu@cs.cmu.edu>

Num of Distance computations *Speedup* for K-NN



collaboration with Ting Liu <tingliu@cs.cmu.edu>

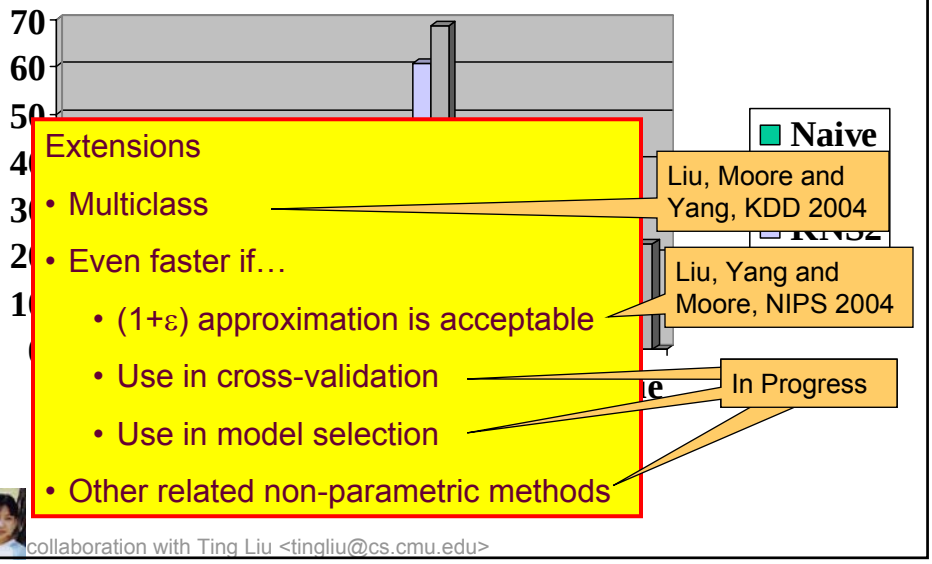
Wall-clock-time speedup for k-NN



collaboration with Ting Liu <tingliu@cs.cmu.edu>

Wall-clock-time speedup for k-NN

Algorithm: Liu, Moore and Gray, NIPS 2003



Extensions

- Multiclass — Liu, Moore and Yang, KDD 2004
- Even faster if…
  - (1+ε) approximation is acceptable — Liu, Yang and Moore, NIPS 2004
  - Use in cross-validation — In Progress
  - Use in model selection
- Other related non-parametric methods

collaboration with Ting Liu <tingliu@cs.cmu.edu>

Outline

Cached Sufficient Statistics

Ball Trees Refresher

K-nearest-neighbor classification (exploiting the question part one)

Non-parametric classification

➤ Biosurveillance and Epidemiology

Scan Statistics (exploiting the question part two)

Bayesian Network Learning

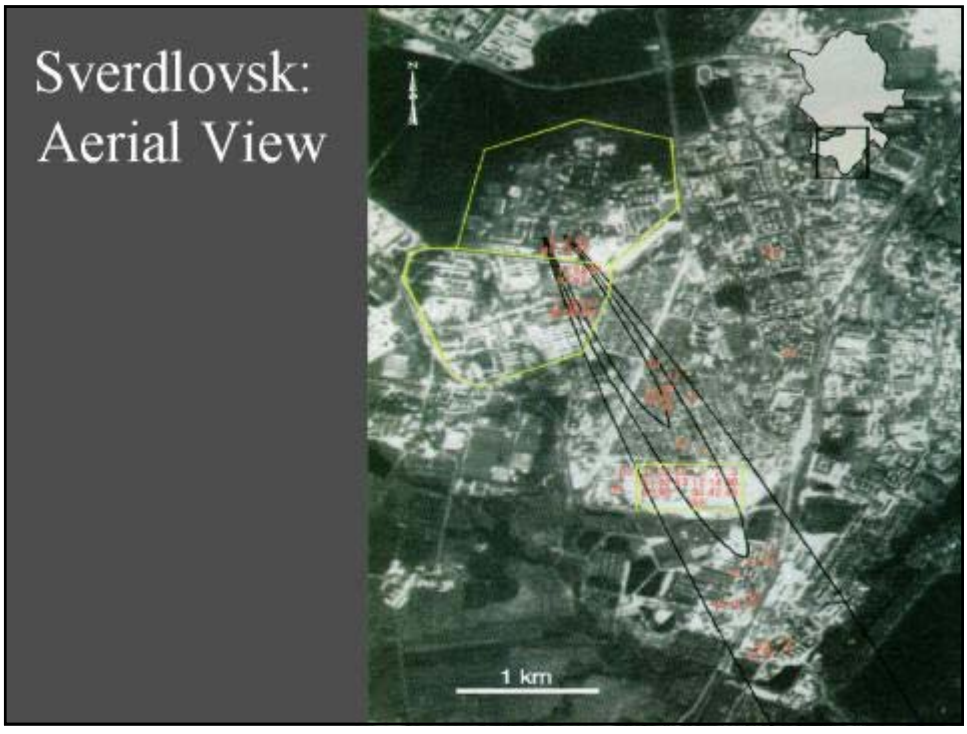
Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

Sensible conclusion

Flaky conclusion

..Early Thursday Morning. Russia. April 1979...

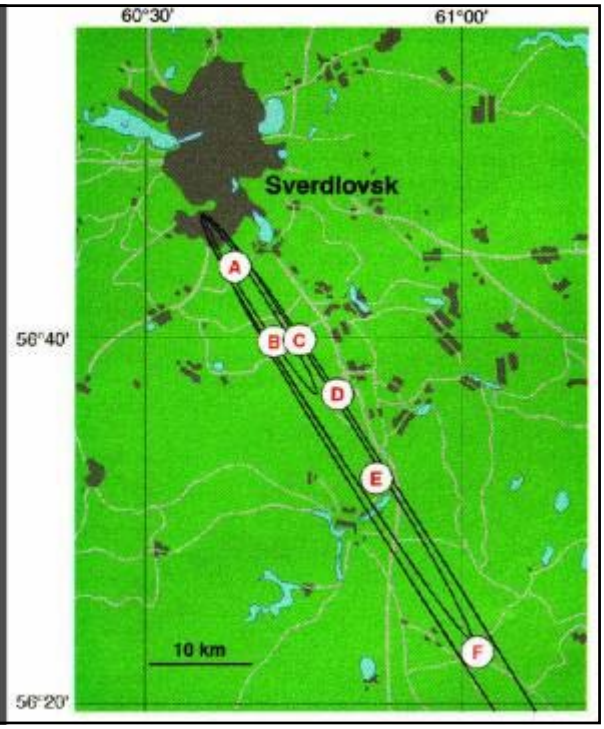
collaboration with Daniel Neill <neill@cs.cmu.edu>



Sverdlovsk: Aerial View

- During April and May 1979, there were 77 Confirmed cases of inhalational anthrax

Sverdlovsk Region: Epi-map



Biosurveillance Algorithms





Biosurveillance Algorithms

Specific Detectors

PANDA2: Patient-based Bayesian Network
[Cooper, Levander et. al]

BARD: Airborne Attack Detection
[Hogan, Cooper]

General Detectors

What's Strange about Recent Events

Fast Scan Statistic
[Neill, Moore]

Fast Scan for Oriented Regions
[Neill, Moore et al.]

Historical Model Scan Statistic
[Hogan, Moore, Neill, Tsui, Wagner]

Bayesian Network Spatial Scan
[Neill, Moore, Schneider, Cooper Wagner, Wong]



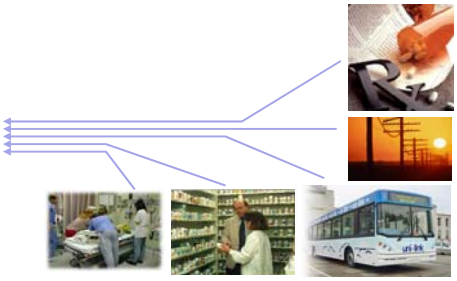


Biosurveillance Algorithms

Specific Detectors

PANDA2: Patient-based Bayesian Network
[Cooper, Levander et. al]

BARD: Airborne Attack Detection
[Hogan, Cooper]



General Detectors

What's Strange about Recent Events

Fast Scan Statistic
[Neill, Moore]

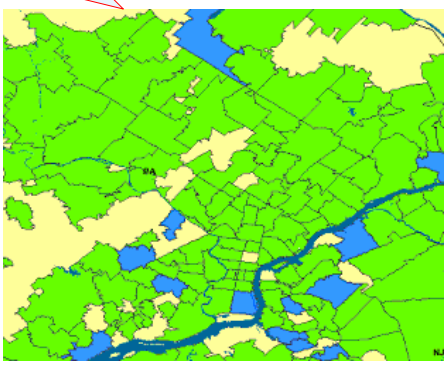
Fast Scan for Oriented Regions
[Neill, Moore et al.]

Historical Model Scan Statistic
[Hogan, Moore, Neill, Tsui, Wagner]

Bayesian Network Spatial Scan
[Neill, Moore, Schneider, Cooper Wagner, Wong]

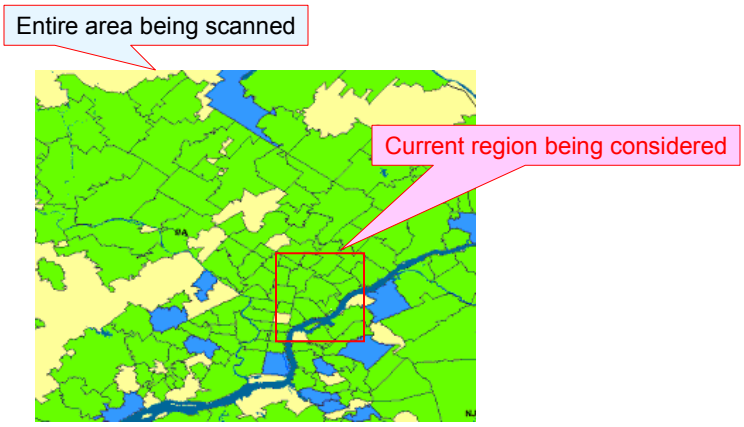
One Step of Spatial Scan

Entire area being scanned



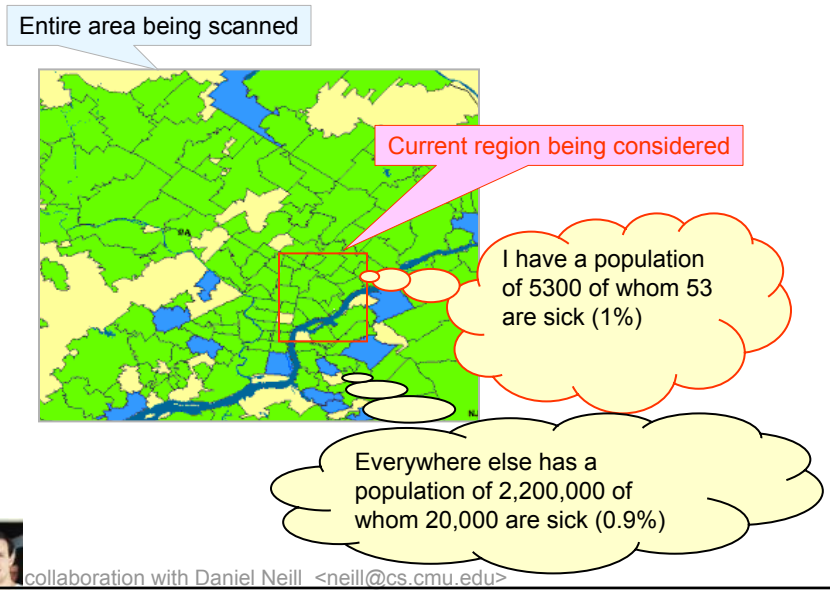
collaboration with Daniel Neill <neill@cs.cmu.edu>

One Step of Spatial Scan



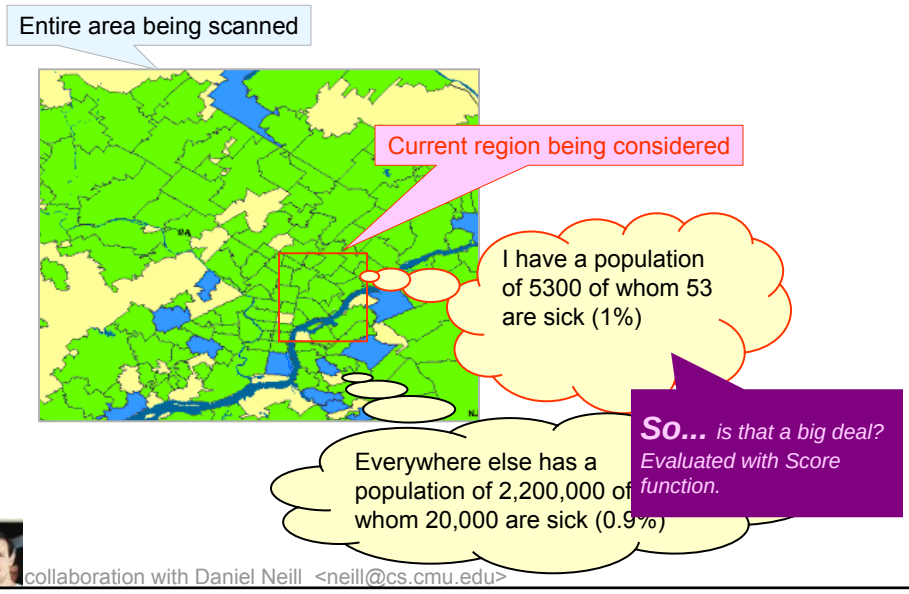
collaboration with Daniel Neill <neill@cs.cmu.edu>

One Step of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

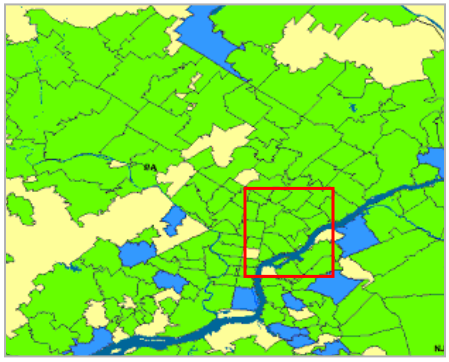
One Step of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

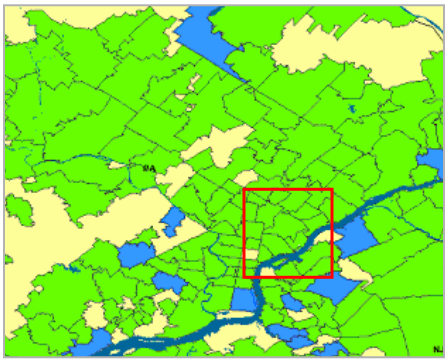
Scoring functions

- Define <u>models</u>:
  - of the null hypothesis H_0: no attacks.
  - of the alternative hypotheses $H_1(S)$: attack in region S.



(Individually Most Powerful statistic for detecting significant increases) *(but still…just an example)*

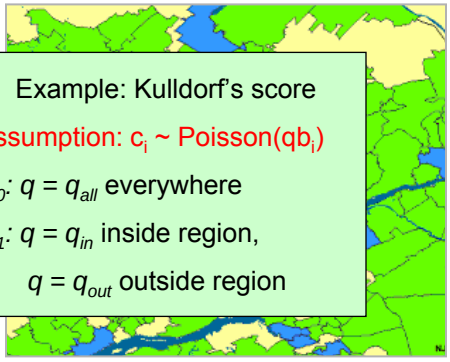
Scoring functions



- Define models:
  - of the null hypothesis H_0: no attacks.
  - of the alternative hypotheses $H_1(S)$: attack in region S.
- Derive a score function *Score(S) = Score(C, B).*
  - Likelihood ratio: $\text{Score}(S) = \frac{L(\text{Data} \mid H_1(S))}{L(\text{Data} \mid H_0)}$
  - To find the most significant region: $S^* = \arg\max_S \text{Score}(S)$

(Individually Most Powerful statistic for detecting significant increases) *(but still…just an example)*

Scoring functions



Example: Kulldorf's score

Assumption: $c_i \sim \text{Poisson}(qb_i)$

H_0: $q = q_{all}$ everywhere

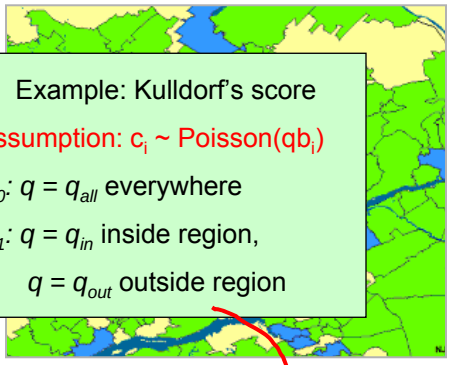
H_1: $q = q_{in}$ inside region,

$q = q_{out}$ outside region

- Define models:
  - of the null hypothesis H_0: no attacks.
  - of the alternative hypotheses $H_1(S)$: attack in region S.
- Derive a score function *Score(S) = Score(C, B).*
  - Likelihood ratio: $\text{Score}(S) = \frac{L(\text{Data} \mid H_1(S))}{L(\text{Data} \mid H_0)}$
  - To find the most significant region: $S^* = \arg\max_S \text{Score}(S)$

(Individually Most Powerful statistic for detecting significant increases) *(but still…just an example)*

Scoring functions



Example: Kulldorf's score

Assumption: $c_i \sim \text{Poisson}(qb_i)$

H_0: $q = q_{all}$ everywhere

H_1: $q = q_{in}$ inside region,

$q = q_{out}$ outside region

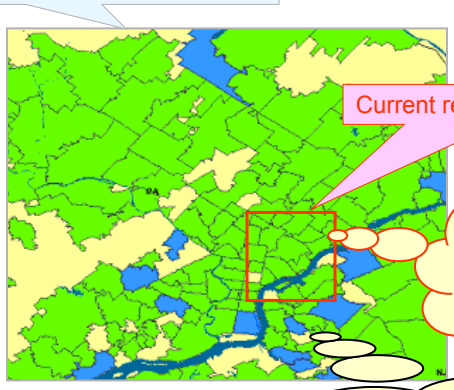
- Define models:
  - of the null hypothesis H_0: no attacks.
  - of the alternative hypotheses $H_1(S)$: attack in region S.
- Derive a score function *Score(S) = Score(C, B).*
  - Likelihood ratio: $\text{Score}(S) = \frac{L(\text{Data} \mid H_1(S))}{L(\text{Data} \mid H_0)}$
  - To find the most significant region: $S^* = \arg\max_S \text{Score}(S)$

$$D(S) = C \log \frac{C}{B} + (C_{tot} - C) \log \frac{C_{tot} - C}{B_{tot} - B} - C_{tot} \log \frac{C_{tot}}{B_{tot}}$$

(Individually Most Powerful statistic for detecting significant increases) *(but still…just an example)*

One Step of Spatial Scan

Entire area being scanned



Current region being considered

I have a population of 5300 of whom 53 are sick (1%)

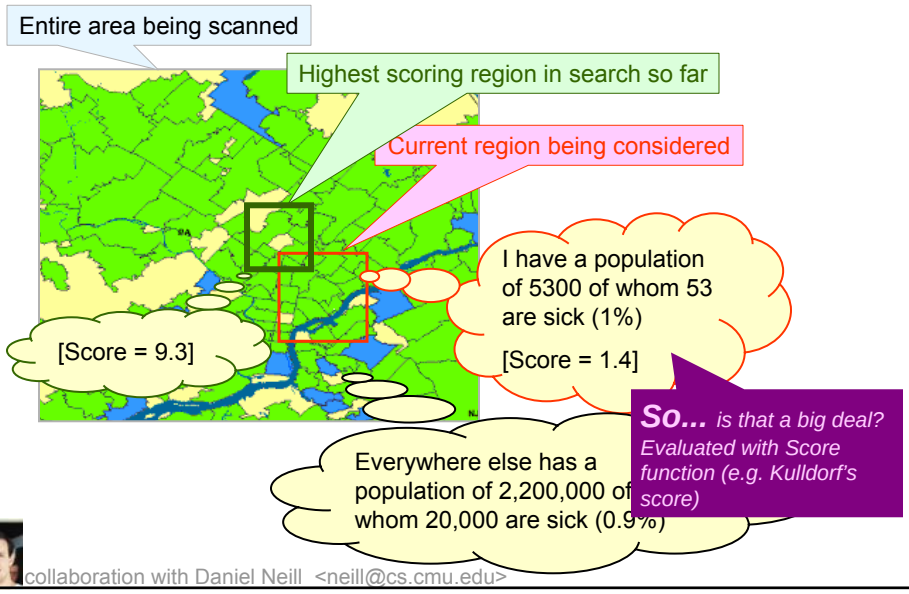
[Score = 1.4]

Everywhere else has a population of 2,200,000 of whom 20,000 are sick (0.9%)

So... *is that a big deal? Evaluated with Score function (e.g. Kulldorf's score)*

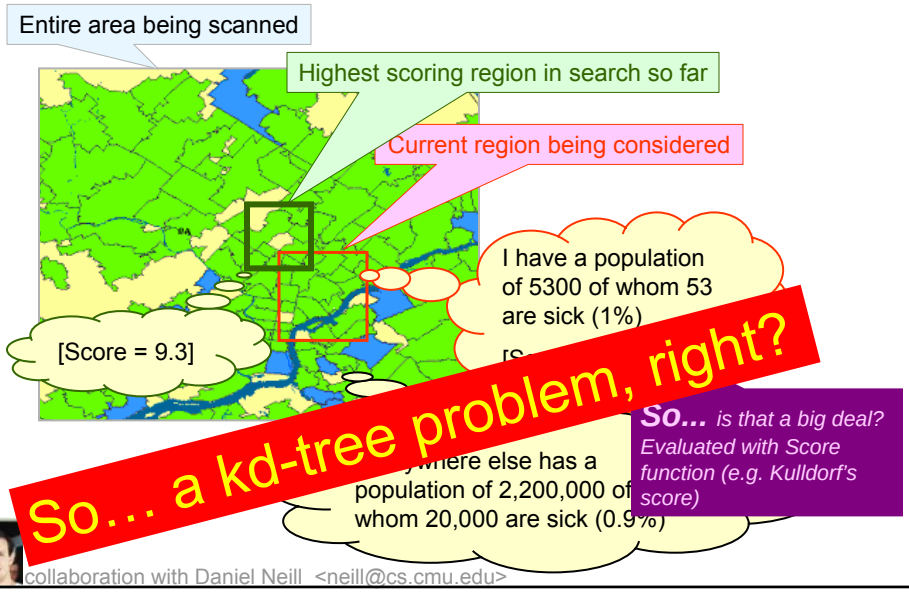
collaboration with Daniel Neill <neill@cs.cmu.edu>

Many Steps of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

Many Steps of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

Computational framework

Data is aggregated to a grid.

B=25 C=27	B=18 C=14	B=22 C=22	B=14 C=15	B=5 C=5
B=25 C=26	B=20 C=17	B=6 C=9	B=20 C=12	B=5 C=4
B=25 C=19	B=25 C=26	B=20 C=43	B=15 C=37	B=20 C=20
B=24 C=18	B=24 C=20	B=19 C=40	B=15 C=32	B=19 C=16
B=23 C=20	B=15 C=17	B=14 C=8	B=10 C=10	B=2 C=3

collaboration with Daniel Neill <neill@cs.cmu.edu>

Computational framework

Data is aggregated to a grid.

Cost of obtaining sufficient statistics for an arbitrary rectangle: O(1)

B=25 C=27	B=18 C=14	B=22 C=22	B=14 C=15	B=5 C=5
B=25 C=26	B=20 C=17	B=6 C=9	B=20 C=12	B=5 C=4
B=25 C=19	B=25 C=26	B=20 C=43	B=15 C=37	B=20 C=20
B=24 C=18	B=24 C=20	B=19 C=40	B=15 C=32	B=19 C=16
B=23 C=20	B=15 C=17	B=14 C=8	B=10 C=10	B=2 C=3

collaboration with Daniel Neill <neill@cs.cmu.edu>

Computational framework

Data is aggregated to a grid.

Cost of obtaining sufficient statistics for an arbitrary rectangle: O(1)

n x n grid has

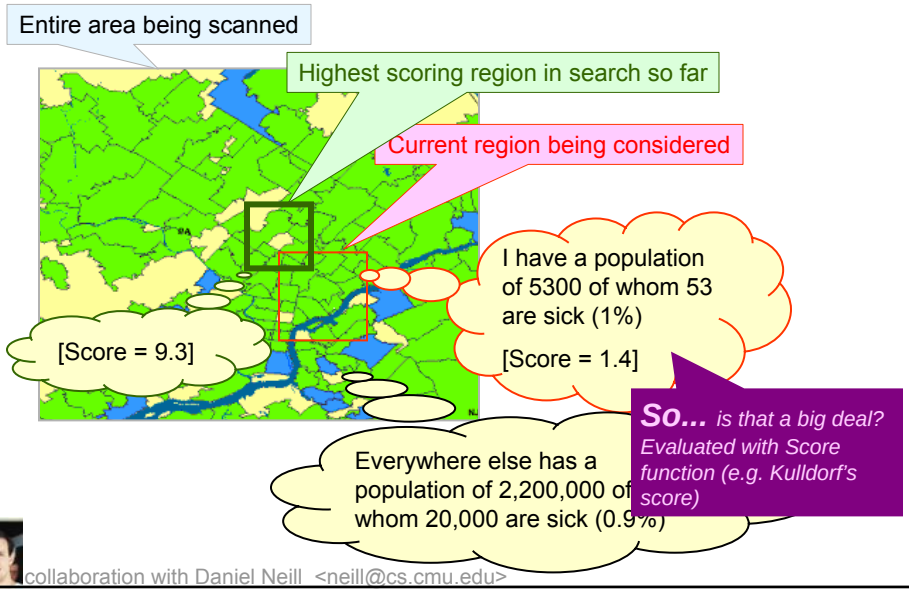
$$\left[\binom{n+1}{2}\right]^2 = O(n^4)$$

rectangles to search

B=25 C=27	B=18 C=14	B=22 C=22	B=14 C=15	B=5 C=5
B=25 C=26	B=20 C=17	B=6 C=9	B=20 C=12	B=5 C=4
B=25 C=19	B=25 C=26	B=20 C=43	B=15 C=37	B=20 C=20
B=24 C=18	B=24 C=20	B=19 C=40	B=15 C=32	B=19 C=16
B=23 C=20	B=15 C=17	B=14 C=8	B=10 C=10	B=2 C=3

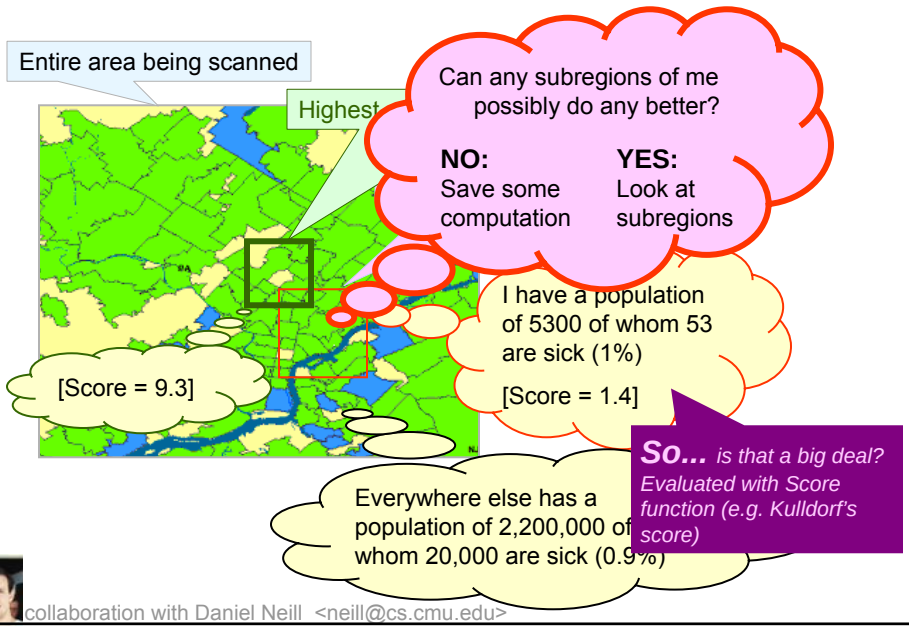
collaboration with Daniel Neill <neill@cs.cmu.edu>

Many Steps of Spatial Scan



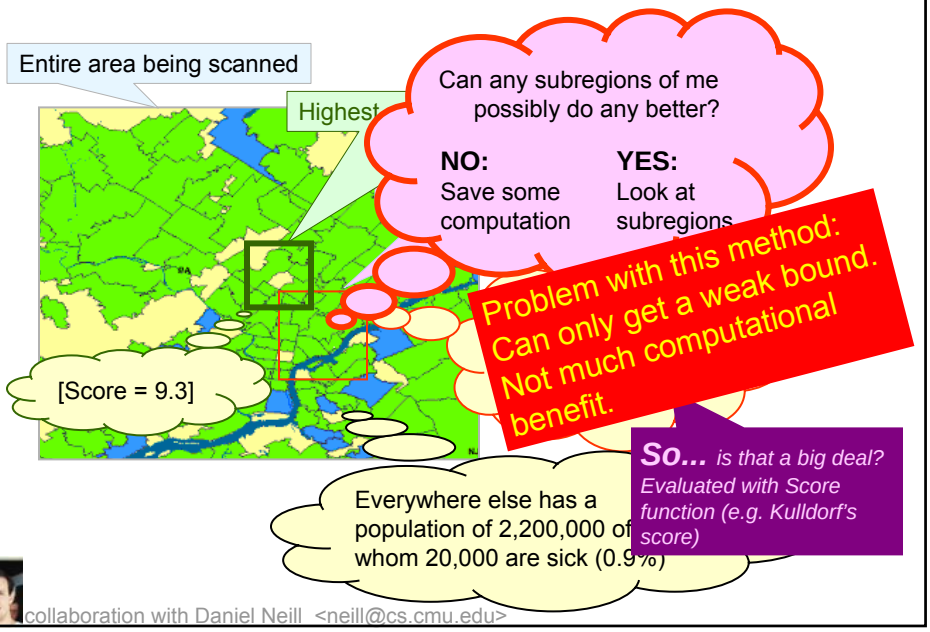
collaboration with Daniel Neill <neill@cs.cmu.edu>

Many Steps of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

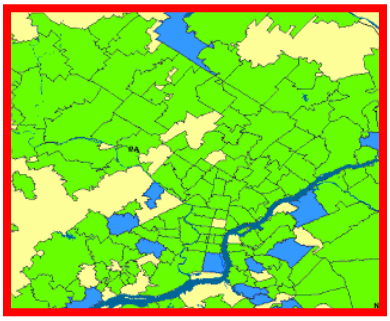
Many Steps of Spatial Scan



collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

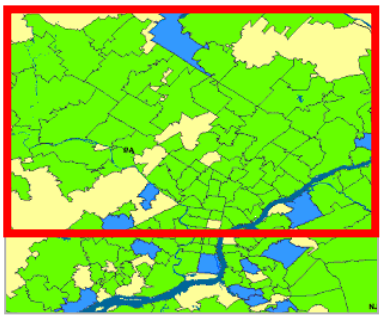


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

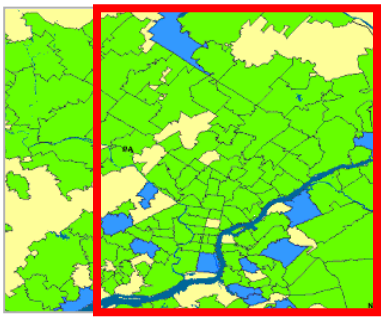


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

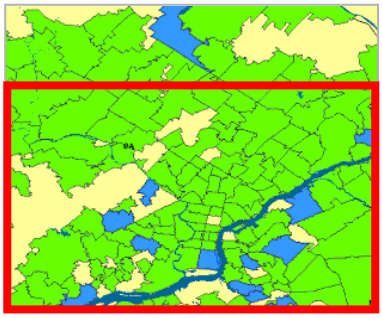


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

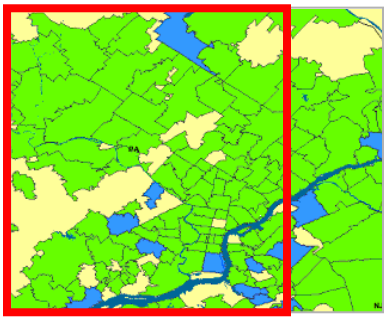


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

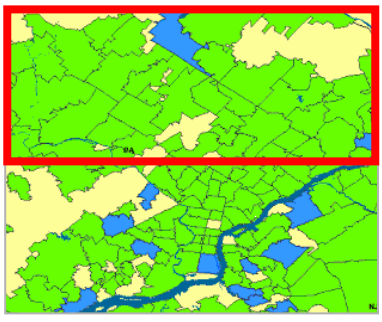


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

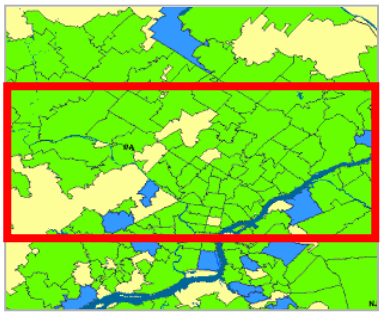


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

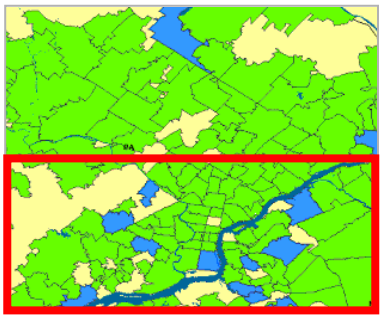


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

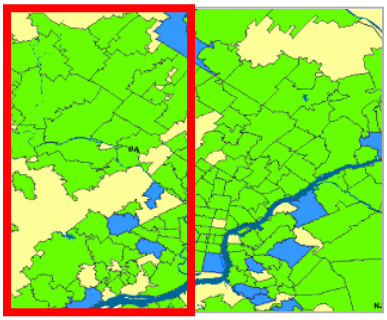


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

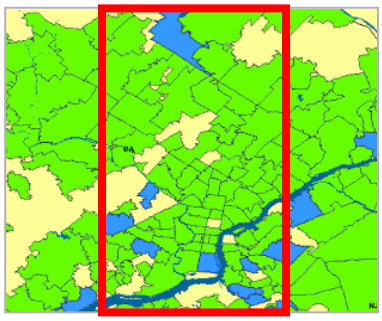


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

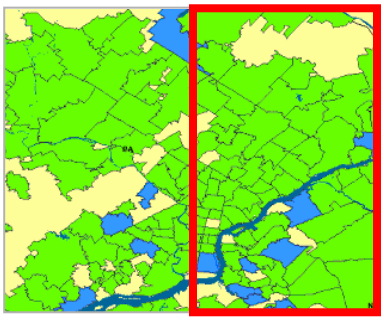


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

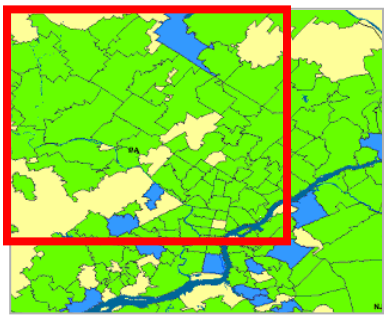


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

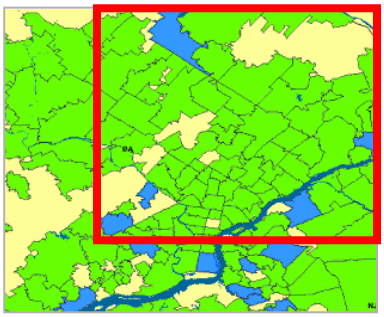


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

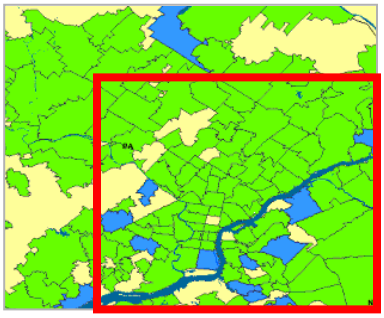


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded

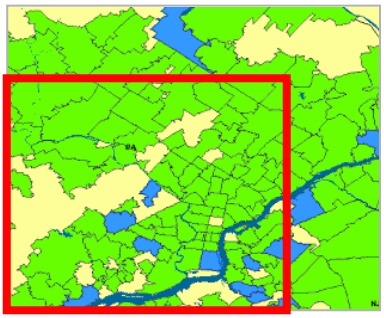


Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

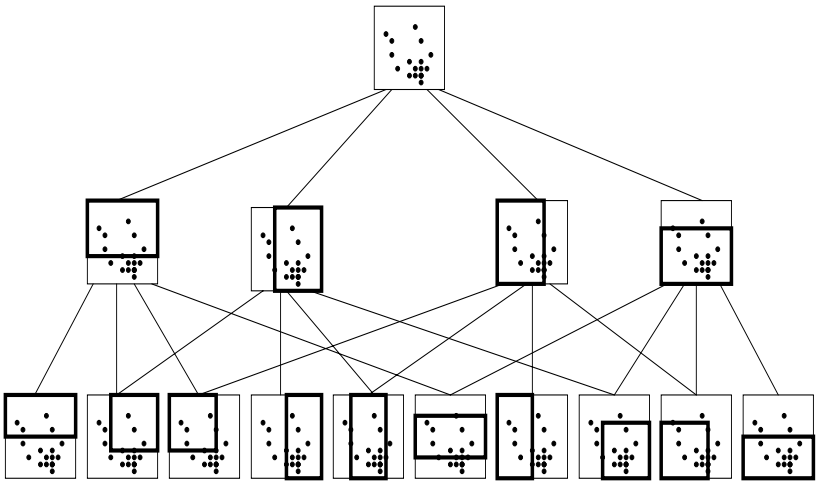
Step 1: Gridded



Check a specific recursive overlapping set of regions called "Gridded Regions"

collaboration with Daniel Neill <neill@cs.cmu.edu>

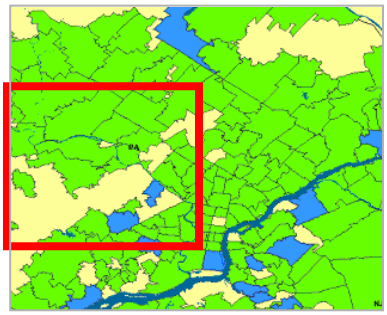
The multi-resolution tree for rectangular regions



collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

Step 1: Gridded



Check a specific recursive overlapping set of regions called "Gridded Regions"

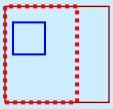
Step 2: Exhaustive

Consider the set of subregions of a Gridded Region.

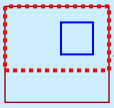
collaboration with Daniel Neill <neill@cs.cmu.edu>

Gridded then Exhaustive

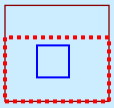
A subregion of me could be one of five types...



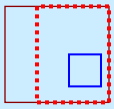
...entirely inside my left gridded child



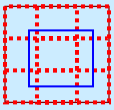
...entirely inside my top gridded child



...entirely inside my bottom gridded child



...entirely in my right gridded child

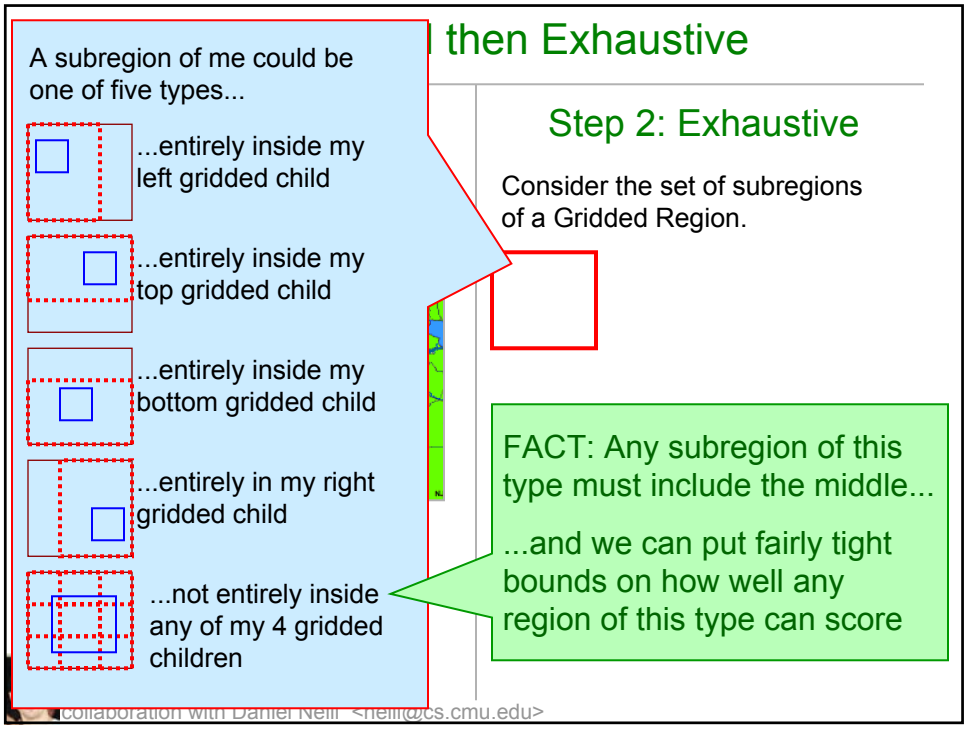


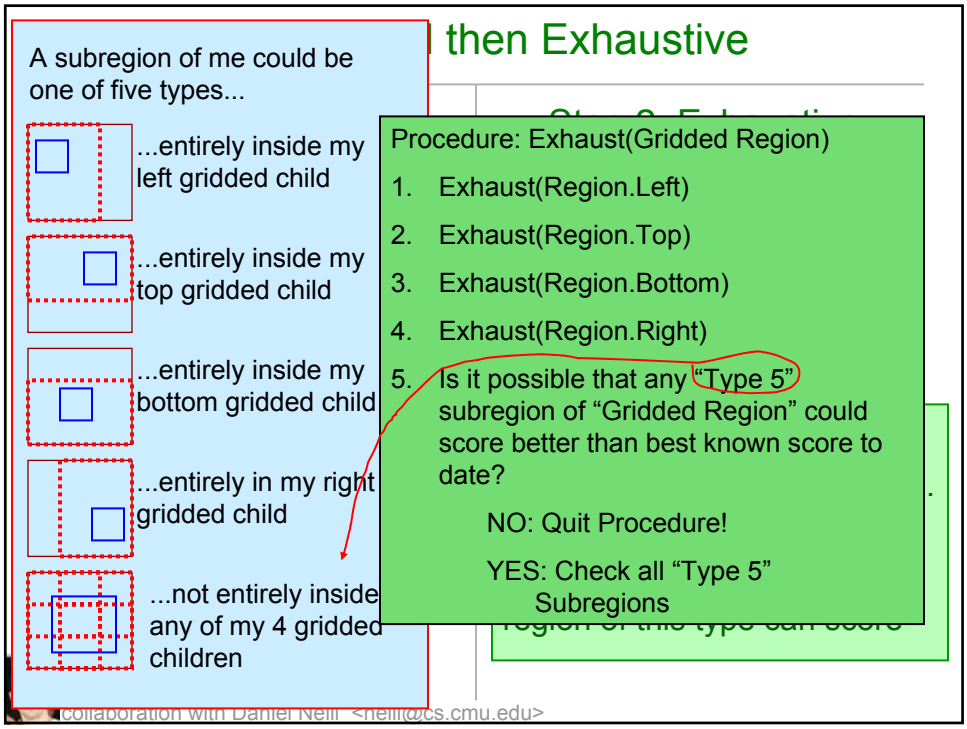
...not entirely inside any of my 4 gridded children

Step 2: Exhaustive

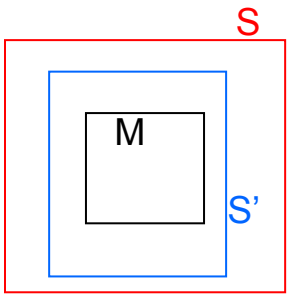
Consider the set of subregions of a Gridded Region.

collaboration with Daniel Neill <neill@cs.cmu.edu>


then Exhaustive
A subregion of me could be one of five types...
...entirely inside my left gridded child
...entirely inside my top gridded child
...entirely inside my bottom gridded child
...entirely in my right gridded child
...not entirely inside any of my 4 gridded children
Step 2: Exhaustive
Consider the set of subregions of a Gridded Region.
FACT: Any subregion of this type must include the middle...
...and we can put fairly tight bounds on how well any region of this type can score
collaboration with Daniel Neill <neill@cs.cmu.edu>


then Exhaustive
A subregion of me could be one of five types...
...entirely inside my left gridded child
...entirely inside my top gridded child
...entirely inside my bottom gridded child
...entirely in my right gridded child
...not entirely inside any of my 4 gridded children
Procedure: Exhaust(Gridded Region)
1. Exhaust(Region.Left)
2. Exhaust(Region.Top)
3. Exhaust(Region.Bottom)
4. Exhaust(Region.Right)
5. Is it possible that any "Type 5" subregion of "Gridded Region" could score better than best known score to date?
NO: Quit Procedure!
YES: Check all "Type 5" Subregions
collaboration with Daniel Neill <neill@cs.cmu.edu>

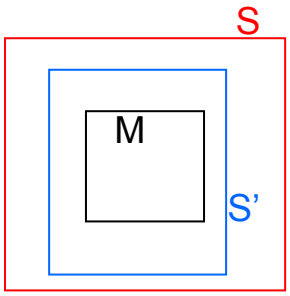
If S' is a middle-containing subregion of S…



5. Is it possible that any "Type 5" subregion of "Gridded Region" could score better than best known score to date?

collaboration with Daniel Neill <neill@cs.cmu.edu>

If S' is a middle-containing subregion of S…

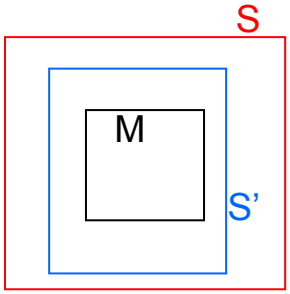


Score(S') = Score(count(S') , baseline(S'))

5. Is it possible that any "Type 5" subregion of "Gridded Region" could score better than best known score to date?

collaboration with Daniel Neill <neill@cs.cmu.edu>

If S' is a middle-containing subregion of S…



An upper bound of c/b for any subregion of S-M

$$d_{inc} \geq \frac{c(S') - c(M)}{b(S') - b(M)}$$

$$b(M) \leq b(S') \leq b(S)$$

$$c(M) \leq c(S') \leq c(S)$$

An upper bound of c/b for any subregion of S that contains M

$$d_{\max} \geq \frac{c(S')}{b(S')}$$

A lower bound on c/b for any subregion of S that excludes M

$$d_{\min} \leq \frac{c(S) - c(S')}{b(S) - b(S')}$$

Score(S') = Score(count(S') , baseline(S'))

5. Is it possible that any "Type 5" subregion of "Gridded Region" could score better than best known score to date?

collaboration with Daniel Neill <neill@cs.cmu.edu>

If S' is a middle-containing subregion of S…

Assume:

$$\frac{\partial}{\partial c}\text{Score}(c,b) \geq 0$$

$$\frac{\partial}{\partial b}\text{Score}(c,b) \leq 0$$

$$\frac{\partial}{\partial b}\text{Score}(c,b) + \frac{c}{b}\frac{\partial}{\partial c}\text{Score}(c,b) \geq 0$$

A lower bound on c/b for any subregion of S that excludes C

$$d_{inc} \geq \frac{c(S') - c(M)}{b(S') - b(M)}$$

$$b(M) \leq b(S') \leq b(S)$$

$$c(M) \leq c(S') \leq c(S)$$

$$d_{\max} \geq \frac{c(S')}{b(S')}$$

$$d_{\min} \leq \frac{c(S) - c(S')}{b(S) - b(S')}$$

Score(S') = Score(count(S') , baseline(S'))

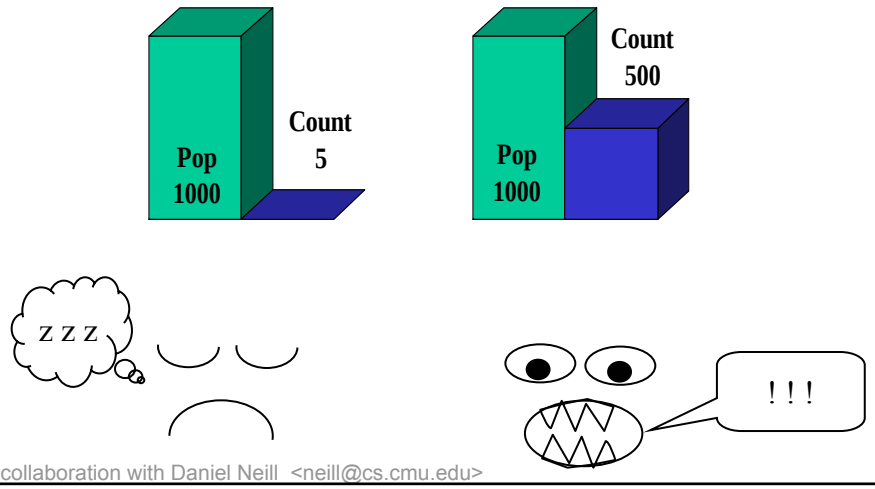
5. Is it possible that any "Type 5" subregion of "Gridded Region" could score better than best known score to date?

collaboration with Daniel Neill <neill@cs.cmu.edu>

Properties of D(S)

$\frac{\partial}{\partial c}\text{Score}(c,b) \geq 0$

Score(S) increases with the total count of S, C(S) = $\sum_S c_i$.

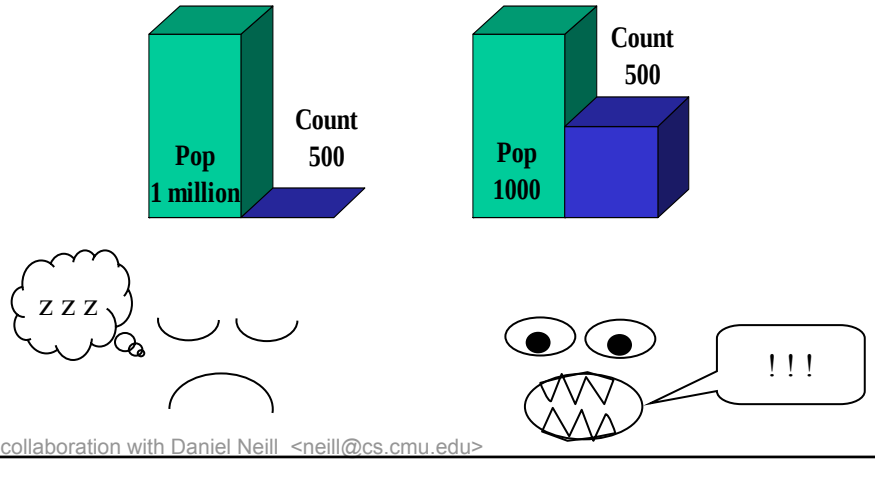


collaboration with Daniel Neill <neill@cs.cmu.edu>

Properties of D(S)

$\frac{\partial}{\partial b}\text{Score}(c,b) \leq 0$

Score(S) decreases with total baseline of S, B(S) = $\sum_S b_i$.

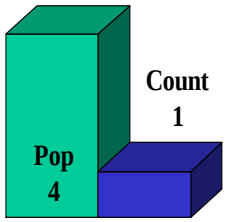


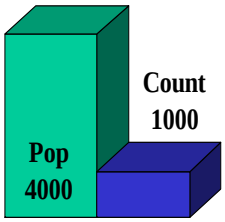
collaboration with Daniel Neill <neill@cs.cmu.edu>

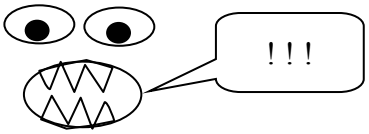
$$\frac{\partial}{\partial b}\text{Score}(c,b)+\frac{c}{b}\frac{\partial}{\partial c}\text{Score}(c,b)\geq 0$$

Properties of D(S)

For a constant ratio C / B, *Score(S)* increases with C and B.







collaboration with Daniel Neill <neill@cs.cmu.edu>

If S' is a middle-containing subregion of S…

Assume:

$$\frac{\partial}{\partial c}\text{Score}(c,b)\geq 0$$

$$\frac{\partial}{\partial b}\text{Score}(c,b)\leq 0$$

$$\frac{\partial}{\partial b}\text{Score}(c,b)+\frac{c}{b}\frac{\partial}{\partial c}\text{Score}(c,b)\geq 0$$

A lower bound on c/b for any subregion of S that excludes C

$$d_{inc}\geq\frac{c(S')-c(M)}{b(S')-b(M)}$$

$$b(M)\leq b(S')\leq b(S)$$

$$c(M)\leq c(S')\leq c(S)$$

$$d_{\max}\geq\frac{c(S')}{b(S')}$$

$$d_{\min}\leq\frac{c(S)-c(S')}{b(S)-b(S')}$$

Score(S') = Score(count(S') , baseline(S'))

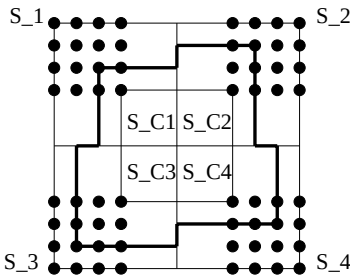
Bottom Line: all the above lets us put a good upper bound on Score(S')

ssible that any "Type 5" subregion
dded Region" could score better
than best known score to date?

collaboration with Daniel Neill <neill@cs.cmu.edu>

Tighter score bounds by quartering

- We precompute global bounds on populations p_{ij} and ratios c_{ij} / p_{ij}, and use these for our initial pruning.
- If we cannot prune the outer regions of S using the global bounds, we do a second pass which is more expensive but allows much more pruning.
- We can use quartering to give much tighter bounds on populations and ratios, and compute a better score bound using these.
  - Requires time quadratic in region size; in effect, we are computing bounds for all irregular but rectangle-like outer regions.



collaboration with Daniel Neill <neill@cs.cmu.edu>

Where are we?

- So we can find the most significant region by searching over the desired set of regions S, and finding the highest D(S).
- Now how can we find whether this region actually is a significant cluster?

collaboration with Daniel Neill <neill@cs.cmu.edu>

Where are we?

- So we can find the most significant region by searching over the desired set of regions S, and finding the highest D(S).
- Now how can we find whether this region actually is a significant cluster?
- Randomization testing

Can sometimes cost us 1000 times more computation!

Though there are further tricks…

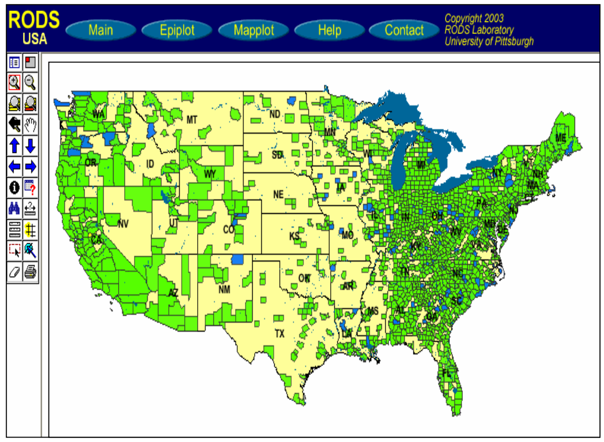
collaboration with Daniel Neill <neill@cs.cmu.edu>

Why the Scan Statistic speed obsession?

collaboration with Daniel Neill <neill@cs.cmu.edu>

Why the Scan Statistic speed obsession?

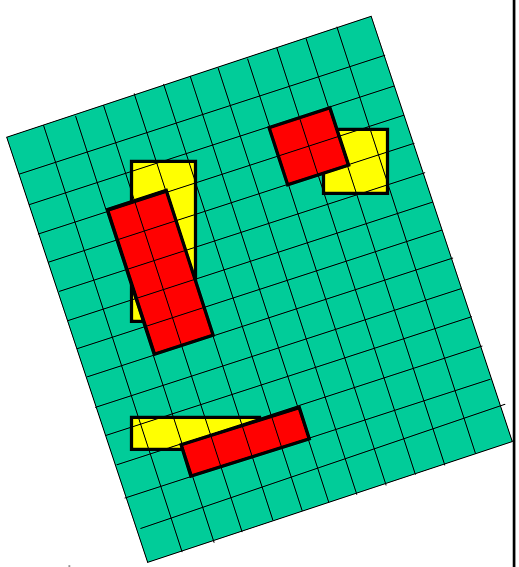
- Traditional Scan Statistics very expensive, especially with Randomization tests
- Going national
- A few hours could actually matter!



Which regions to search?

- We choose to search over the sp... ...ngular regi...
  ers
  es
  ...ology
  ...ers are
  ...ated (e.g. from wind... pathogens).
  - Important in brain imaging because of the brain's "folded sheet" structure.

We can find non-axis-aligned rectangles by examining multiple rotations of the data.



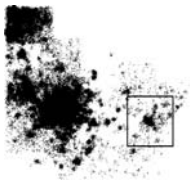
Results

collaboration with Daniel Neill <neill@cs.cmu.edu>

Summary of results

- The fast spatial scan results in huge speedups (as compared to exhaustive search), making fast real-time detection of clusters feasible.
- No loss of accuracy: fast spatial scan finds the exact same regions and p-values as exhaustive search.

OTC data from National Retail Data Monitor



ED data

collaboration with Daniel Neill <neill@cs.cmu.edu>

Performance comparison

Algorithm name	Search space	Number of regions	Search time (total)	Time / region	Likelihood ratio
SaTScan	Circles centered at datapts	150 billion	16 hours	400 ns	413.56
exhaustive	Axis-aligned rectangles	1.1 trillion	45 days	3600 ns	429.85
fast spatial scan	Axis-aligned rectangles	1.1 trillion	81 minutes	4.4 ns	429.85

- On ED dataset (600,000 records), 1000 replicas
- For SaTScan: M=17,000 distinct spatial locations
- For Exhaustive/fast: 256 x 256 grid

collaboration with Daniel Neill <neill@cs.cmu.edu>

Performance comparison

Algorithm name	Search space	Number of regions	Search time (total)	Time / region	Likelihood ratio
SaTScan	Circles centered at datapts	150 billion	16 hours	400 ns	413.56
exhaustive	Axis-aligned rectangles	1.1 trillion	45 days	3600 ns	429.85
fast spatial scan	Axis-aligned rectangles	1.1 trillion	81 minutes	4.4 ns	429.85

- On ED dataset (600,000 records), 1000 replicas
- For SaTScan: M=17,000 distinct spatial locations
- For Exhaustive/fast: 256 x 256 grid

- Algorithms: Neill and Moore, NIPS 2003, KDD 2004
- Deployment: Neill, Moore, Tsui and Wagner, *Journal of Death and Doom*, Nov. '04

collaboration with Daniel Neill <neill@cs.cmu.edu>

Performance comparison

Algorithm name	Search space	Number of regions	Search time (total)	Time / region	Likelihood ratio
SaTScan	Circles centered at datapts	150 billion	16 hours	400 ns	413.56
exhaustive	Axis-aligned rectangles	1.1 trillion	45 days	3600 ns	429.85
fast spatial scan	Axis-aligned rectangles	1.1 trillion	81 minutes	4.4 ns	429.85

- On ED dataset (600,000 records), 1000 replicas
- For SaTScan: M=17,000 distinct spatial locations
- For Exhaustive/fast: 256 x 256 grid

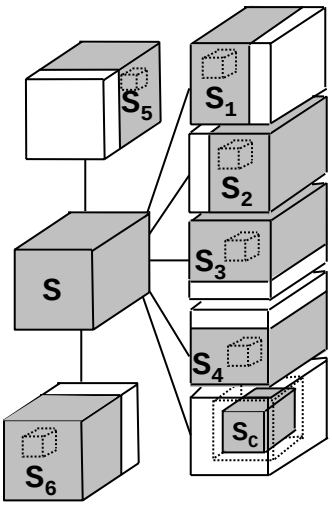
- Algorithms: Neill and Moore, NIPS 2003, KDD 2004
- Deployment: Neill, Moore, Tsui and Wagner, Journal of [illegible], Nov. '04

Morbidity and Mortality Weekly Report

collaboration with Daniel Neill <neill@cs.cmu.edu>

d-dimensional partitioning

- Parent region S is divided into 2d overlapping children: an "upper child" and a "lower child" in each dimension.
- Then for any rectangular subregion S' of S, exactly one of the following is true:
  - S' is contained entirely in (at least) one of the children $S_1 \ldots S_{2d}$.
  - S' contains the center region S_C, which is common to all the children.
- Starting with the entire grid G and repeating this partitioning recursively, we obtain the overlap-kd tree structure.

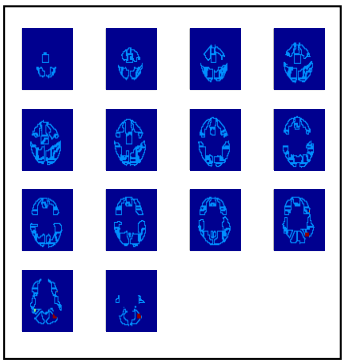


- Algorithm: Neill, Moore and Mitchell NIPS 2004

collaboration with Daniel Neill <neill@cs.cmu.edu>

Results: OTC, fMRI

- fMRI data (64 x 64 x 14 grid):
  - 7-148x speedups as compared to exhaustive search approach.



fMRI data from noun/verb word recognition task

collaboration with Daniel Neill <neill@cs.cmu.edu>

Limitations of the algorithm

- Data must be aggregated to a grid.
- Not appropriate for very high-dimensional data.
- Assumes that we are interested in finding (rotated) rectangular regions.
- Less useful for special cases (e.g. square regions, small regions only).
- Slower for finding multiple regions.

collaboration with Daniel Neill <neill@cs.cmu.edu>

Related work (1)

- Machine learning / data mining: general cluster detection approaches (e.g. bump hunting, CLIQUE, DBSCAN).
  - Generally, heuristic aggregation of individual high-density cells; clusters found are not “optimal” in any well-defined sense.
  - No conclusions can be drawn about significance of clusters; does not answer the question of which clusters are “real” ones and which are likely due to chance.
  - No underlying model (thus can’t adapt to different domains with different models).

collaboration with Daniel Neill <neill@cs.cmu.edu>

Related work (2)

- Brain imaging: application-specific cluster detection approaches (e.g. Statistical Parametric Mapping, Worsley et al.).
  - Generally, tests significance on a per-voxel basis after applying some method of spatial smoothing.
  - Clusters must be inferred by grouping individually significant voxels; no per-cluster false positive rate is guaranteed.
  - Assumes Gaussian Random Field: Not easily applicable to other domains with other underlying models.

collaboration with Daniel Neill <neill@cs.cmu.edu>

Related work (3)

- Epidemiology: four main areas.
  - non-specific clustering: evaluates general tendency of data to cluster
  - focused clustering: evaluates risk w.r.t. a given spatial location (e.g. potential hazard)
  - disease mapping: models spatial variation in risk by applying spatial smoothing.
  - spatial scan statistics (and related techniques).

collaboration with Daniel Neill <neill@cs.cmu.edu>

Outline

Cached Sufficient Statistics

Ball Trees Refresher

K-nearest-neighbor classification (exploiting the question part one)

Non-parametric classification

Biosurveillance and Epidemiology

Scan Statistics (exploiting the question part two)

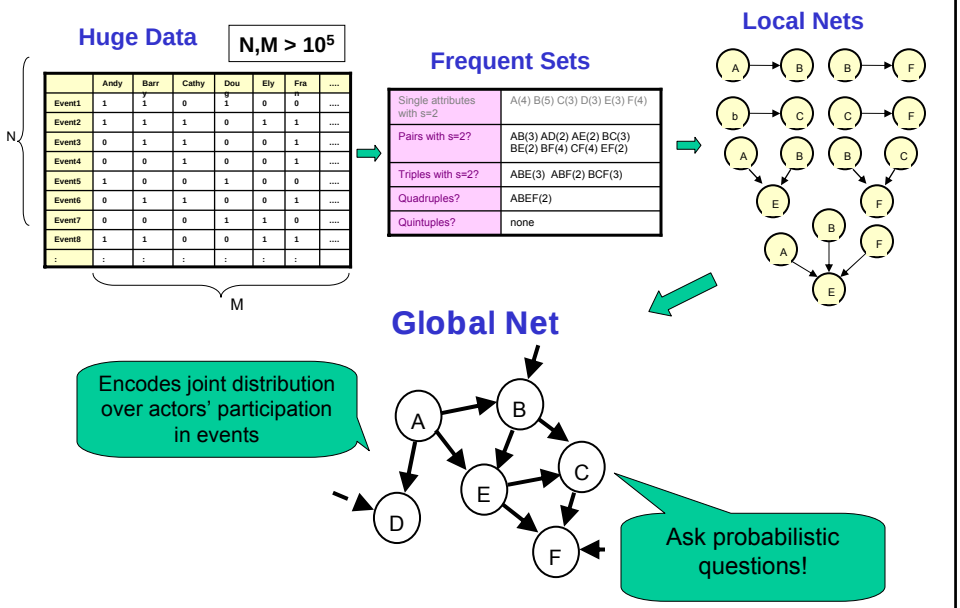
Bayesian Network Learning

Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

Sensible conclusion

Flaky conclusion

SBNS



Massive Sparse Data

- Social Nets
- Transactional Data (market basket)
- Weblogs
- Warehousing

Data is massive and sparse

	Andy	Barry	Cathy	Doug	Ely	Fran	….
Record1	1	1	0	1	0	0	….
Record2	1	1	0	0	1	0	….
Record3	0	1	1	0	0	1	….
Record4	0	0	1	0	0	1	….
Record5	1	0	0	1	0	0	….
Record6	0	1	1	0	0	1	….
Record7	0	0	0	1	1	0	….
Record8	1	1	0	0	1	1	….
:	:	:	:	:	:	:	

N

M

$N,M > 10^5$

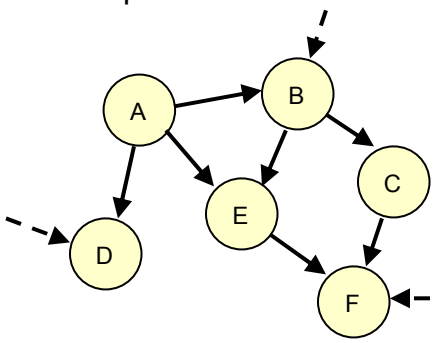
Records are events: meetings, co-authorship, items purchased together, user preferences (tv shows), protein co-activation etc

Bayes Net Structure Search

Input: A Dataset

	Andy	Barry	Cathy	Doug	Ely	Fran	….
Record 1	1	1	0	1	0	0	….
Record 2	1	1	0	0	1	0	….
Record 3	0	1	1	0	0	1	….
Record 4	0	0	1	0	0	1	….
Record 5	1	0	0	1	0	0	….
Record 6	0	1	1	0	0	1	….
Record 7	0	0	0	1	1	0	….
Record 8	1	1	0	0	1	1	….
:	:	:	:	:	:	:	

Output: A DAG + CPTs



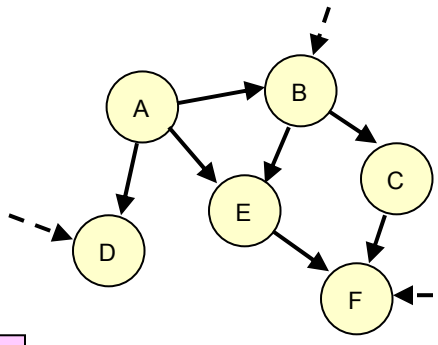
Goal: A DAG that best explains the data

Bayes Net Structure Search

Input: A Dataset

	Andy	Barry	Cathy	Doug	Ely	Fran	....
Record 1	1	1	0	1	0	0	....
Record 2	1	1	0	0	1	0	....
Record 3	0	1	1	0	0	1	....
Record 4	0	0	1	0	0	1	....
Record 5	1	0	0	1	0	0	....
Record 6	0	1	1	0	0	1	....
Record 7	0	0	0	1	1	0	....
Record 8	1	1	0	0	1	1	....
:	:	:	:	:	:	:	

Output: A DAG + CPTs



Computational Goal: A DAG that scores best using a scoring function

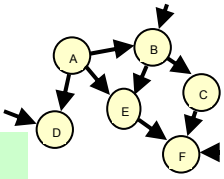
Goal: A DAG that best explains the data

Structure Scoring

Input: A Dataset

	Andy	Barry	Cathy	Doug	Ely	Fran	....
Record1	1	1	0	1	0	0	....
Record2	1	1	0	0	1	0	....
Record3	0	1	1	0	0	1	....
Record4	0	0	1	0	0	1	....
Record5	1	0	0	1	0	0	....
Record6	0	1	1	0	0	1	....
Record7	0	0	0	1	1	0	....
Record8	1	1	0	0	1	1	....
:	:	:	:	:	:	:	

Output: A DAG + CPTs



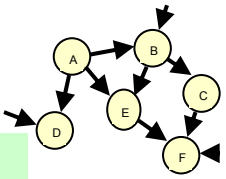
$$\text{Score}(\text{DAG}) = \sum_{n \in \text{Nodes}} \text{Nodescore}(\text{Parents}_n \rightarrow n)$$

Structure Scoring

Input: A Dataset

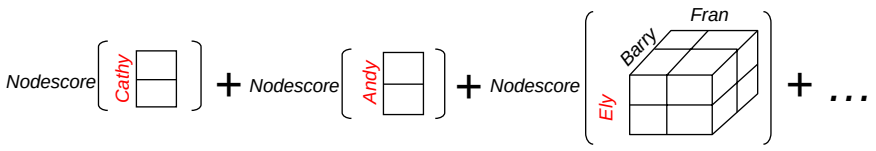
	Andy	Barry	Cathy	Doug	Ely	Fran	....
Record1	1	1	0	1	0	0	....
Record2	1	1	0	0	1	0	....
Record3	0	1	1	0	0	1	....
Record4	0	0	1	0	0	1	....
Record5	1	0	0	1	0	0	....
Record6	0	1	1	0	0	1	....
Record7	0	0	0	1	1	0	....
Record8	1	1	0	0	1	1	....
:	:	:	:	:	:	:	

Output: A DAG + CPTs



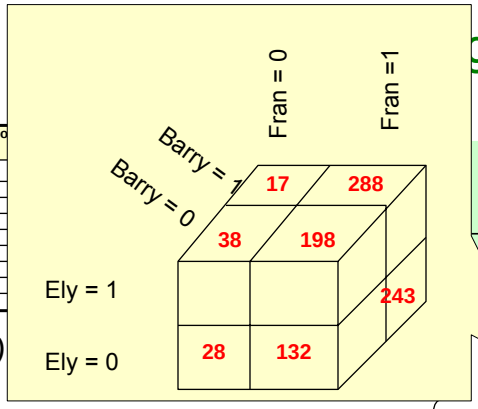
$$\text{Score}(\text{DAG}) = \sum_{n \in \text{Nodes}} \text{Nodescore}(\text{Parents}_n \rightarrow n)$$

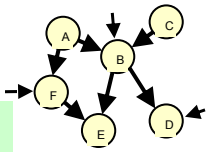
Score(DAG) =



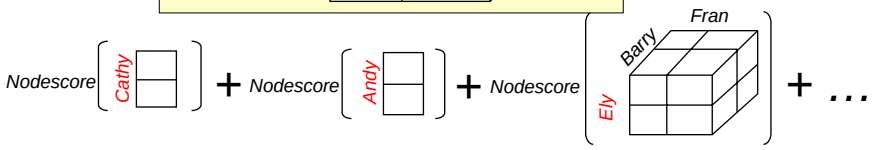
Input: A Dataset

	Andy	Barry	Cathy	Do
Record1	1	1	0	1
Record2	1	1	0	0
Record3	0	1	1	0
Record4	0	0	1	0
Record5	1	0	0	1
Record6	0	1	1	0
Record7	0	0	0	1
Record8	1	1	0	0
:	:	:	:	:





Score(DAG)



Hill Climbing Problems

- Local Optima
- $O(M^2R + KMR)$ time until converge

K = # iterations

Hill Climbing Solutions

Help / Hinder

- Multi-Restart
- Simulated Annealing
- TABU search
- Multi-step lookahead

Hill Climbing Problems

- Local Optima
- $O(M^2R + KMR)$ time until converge

Too Expensive!!!

Hill Climbing Solutions

Help / Hinder

- Multi-Restart
- Simulated Annealing
- TABU search
- Multi-step lookahead

Sparsness

	Andy	Barry	Cathy	Doug	Ely	Fran	….
Record1	1	1	0	1	0	0	….
Record2	1	1	0	0	1	0	….
Record3	0	1	1	0	0	1	….
Record4	0	0	1	0	0	1	….
Record5	1	0	0	1	0	0	….
Record6	0	1	1	0	0	0	
Record7	0	0	0	1	1		
Record8	1	1	0	0	1		
:	:	:	:	:	:		

More useful than just for count caching!

Property: SPARSNESS => Frequent Sets

Frequent Sets

- Which **pairs** happen with frequency >=s (e.g. s=2)? (*s* aka support)

(Andy, Barry), (Andy, Doug), (Andy, Ely), (Barry, Cathy), (Barry, Ely), (Barry, Fran), (Cathy, Fran)

	Andy	Barry	Cathy	Doug	Ely	Fran	….
Record1	1	1	0	1	0	0	….
Record2	1	1	0	0	1	0	….
Record3	0	1	1	0	0	1	….
Record4	0	0	1	0	0	1	….
Record5	1	0	0	1	0	0	….
Record6	0	1	1	0	0	1	….
Record7	0	0	0	1	1	0	….
Record8	1	1	0	0	1	1	….
:	:	:	:	:	:	:	

Frequent Sets

- Which triples happen with frequency >= s *(s=2)*?

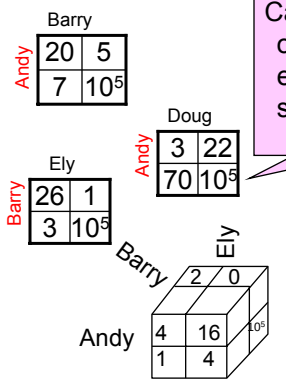
  (Andy, Barry, Ely) (Barry, Cathy, Fran)

	Andy	Barry	Cathy	Doug	Ely	Fran	….
Record1	1	1	0	1	0	0	….
Record2	1	1	0	0	1	0	….
Record3	0	1	1	0	0	1	….
Record4	0	0	1	0	0	1	….
Record5	1	0	0	1	0	0	….
Record6	0	1	1	0	0	1	….
Record7	0	0	0	1	1	0	….
Record8	1	1	0	0	1	1	….
:	:	:	:	:	:	:	

SBNS Step 1

- Find all frequent sets FS and store counts:
  count(FS) > s, |FS| < max_tuple_size

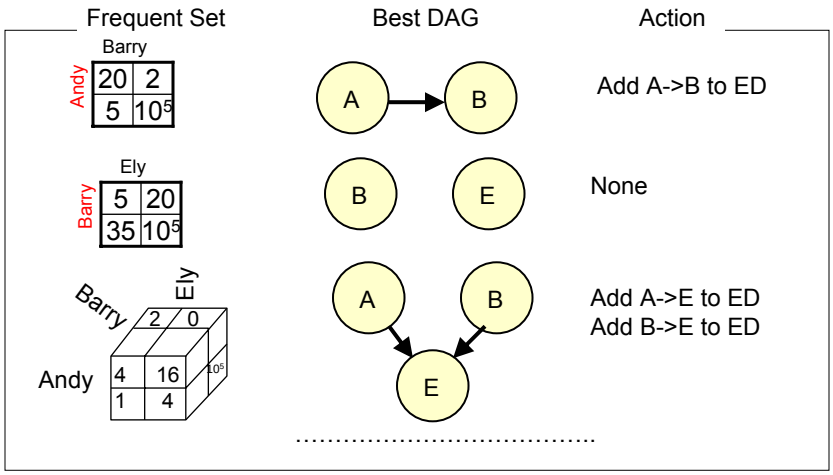
(Andy, Barry),
(Andy, Doug),
(Andy, Ely),
(Barry, Cathy),
(Barry, Ely),
:
(Andy, Barry, Ely)
(Barry, Cathy, Fran)



Can use ADTrees or other DS for efficient count storage

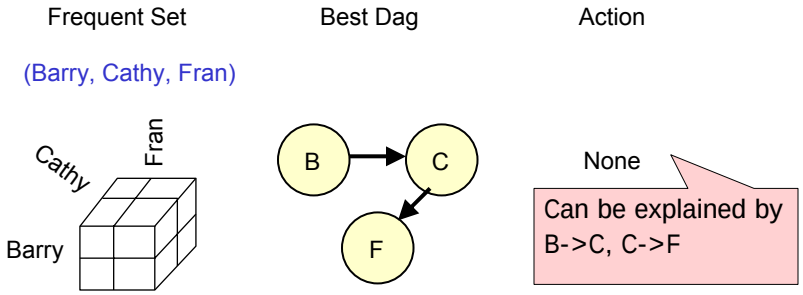
SBNS Step 2

Find Best Local Dags (using BDeu):



SBNS Step 2 cont'd

Find Best Local Dags (using BDeu):



Correlation

Simple case: Pairs

POSITIVE

x \ y	v	kv
kv	N-2kv-v	

$v << N$

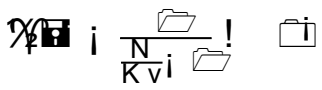
$k -> 0$

$$\hat{\rho} = \frac{1}{1+k} - \frac{kv}{N-kv-v} \underset{k->0}{\rightarrow} 1$$

NEGATIVE

x \ y	0	Kv
Kv	N-2Kv	

$Kv << N$



Correlation

POSITIVE

x \ y	v	kv
kv	N-2kv-v	

$v << N$

$k -> 0$

$$\hat{\rho} = \frac{1}{1+k} - \frac{kv}{N-kv-v} \underset{k->0}{\rightarrow} 1$$

NEGATIVE

x \ y	0	Kv
Kv	N-2Kv	

$Kv << N$

Frequent Sets

Frequent Set with support s - set of attributes that co-occur s or more times

- Data mining concept
- Efficient algorithms to get counts from sparse data

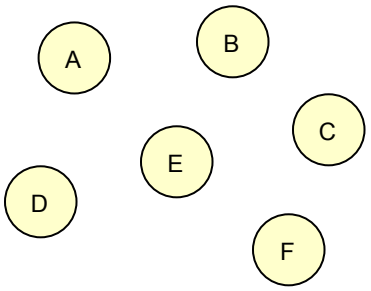
Visiting Edgedump:

Edges	Number of Occurrences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:

Note: The edges are added in the order of their importance

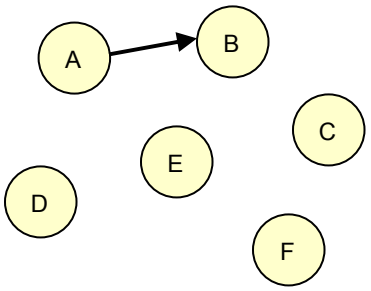
Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:

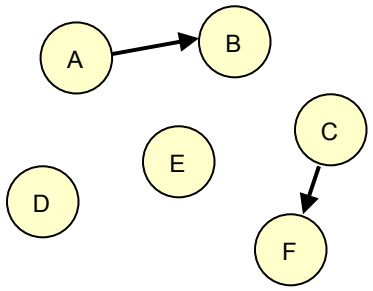


Score(B|par(B)=Ø) = 5.2

Score(B|par(B)=A) = 7.1

Step 3: Creating global Bayes Net

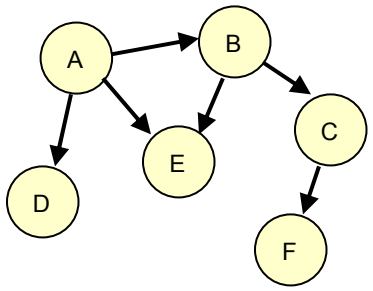
Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



Score(F|par(F)=Ø) = 4.3
Score(F|par(F)=C) = 5.2

Step 3: Creating global Bayes Net

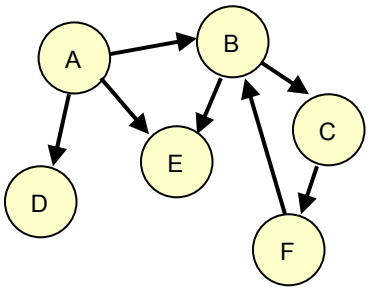
Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



Score(E|par(E)=A) = 2.7
Score(E|par(E)={A,B}) = 6.1

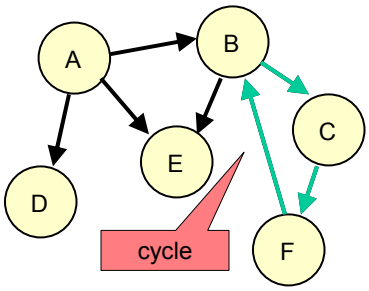
Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



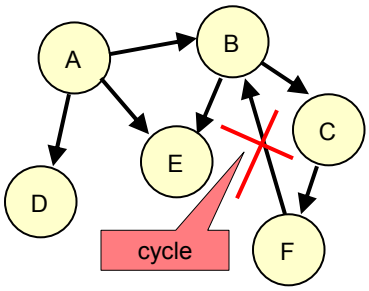
Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



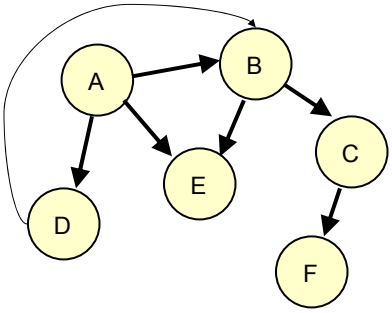
Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:

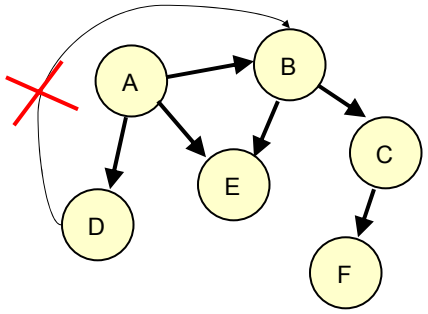


Score(B|par(B)=A) = 5.2

Score(B|par(B)={A,D}) = 4.3

Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:

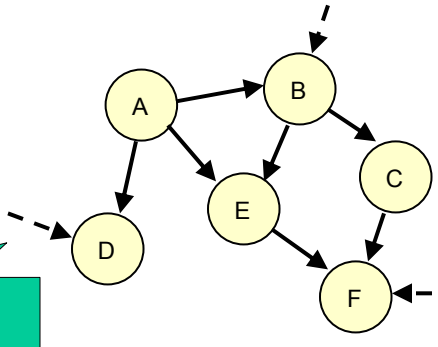


Score(B|par(B)=A) = 5.2

Score(B|par(B)={A,D}) = 4.3

Step 3: Creating global Bayes Net

Edges	Number of Occurences in DAGs
Andy -> Barry	6
Cathy -> Fran	4
Andy -> Doug	3
Andy -> Ely	3
Barry -> Cathy	2
Barry -> Ely	1
Fran -> Barry	1
Doug -> Barry	1
Ely -> Fran	1
:	:



Best Scoring DAG

Negative Correlations

- Remember example:

	y	
x	0	Kv
	Kv	N-2Kv

$Kv << N$

Negative Correlations

x and *y*: $N_x > s$, $N_y > s$, $N_{xy} = \mathbf{0}$,

Mutual Information:

$I_{xy} = f(N, N_x, N_y)$,

f – monotonically increasing with N_x *and* N_y

(Meila, 1999)

Negative Correlations

x and y: $N_x > s$, $N_y > s$, $N_{xy} = \mathbf{0}$,

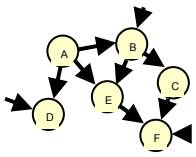
Mutual Information:

$I_{xy} = f(N, N_x, N_y)$,

f – monotonically increasing with N_x *and* N_y (Meila, 1999)

Interested in frequencies with highest counts!

MI Augmentation



Descending frequencies

Descending frequencies

E → F B … C A D …….

F → B … C A D …….

B → … C A D ……

: : :

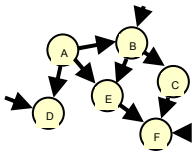
C → A D …….

A → D …….

D → …….

…

MI Augmentation



Co-occurred

Descending frequencies

Descending frequencies

E → ~~F~~ ~~B~~ … C ~~A~~ D …….

F → ~~B~~ … ~~C~~ A D …….

B → … ~~C~~ ~~A~~ D ……

: : :

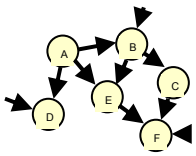
C → A D …….

A → ~~D~~…….

D → …….

…

MI Augmentation



Descending frequencies

Descending frequencies

E → F B … C A D …….

F → B … C A D …….

B → … C A D ……

: : :

C → A D …….

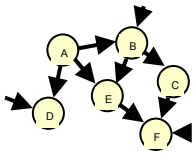
A → D …….

D → …….

…

O(M²) ?

MI Augmentation Trick



Descending frequencies

Descending frequencies

E → F B … C A D …….

F → B … C A D …….

B → … C A D ……

: : :

C → A D …….

A → D …….

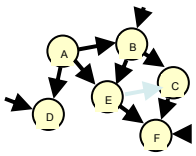
D → …….

…

E -> C Increases Score?
Yes! -> Add

E -> D Increases Score?
No! -> Prune!

MI Augmentation Trick



Descending frequencies

Descending frequencies

E → F B … C A D …….

F → B … C A D …….

B → … C A D ……

: : :

C → A D …….

A → D …….

D → …….

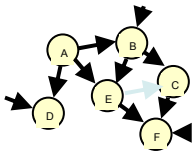
…

Pruned!

E -> C Increases Score?
Yes! -> Add

E -> D Increases Score?
No! -> Prune!

MI Augmentation Trick



Descending frequencies

Descending frequencies

E → F B … C A D ……

F → B … C A D ……

B → … C A D ……

: : :

C → A D …….

A → D …….

D → …….

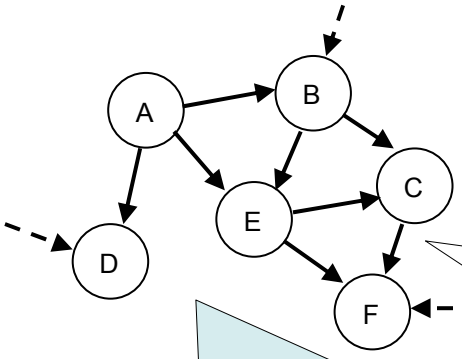
…

Pruned!

F -> B? Cycle!
B -> F? Increases Score?
No! -> Prune!

In practice
~~$O(M^2)$~~

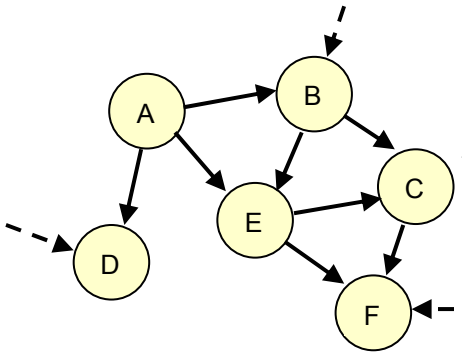
Result



Reduced noise
Positive and Negative correlations

To improve score:
- Hillclimbing
- 2nd Degree Edge edges (parent(x)->child(x))

Result



Reduced noise
Positive and Negative correlations

Datasets

Datasets	Entities	Records
Institute	456	1488
Drinks	136	4744
IMDB	100717	49298
Citeseer	104801	180395

Empirical Results

2003: Largest Bayes Net learning algorithm (Microsoft) had achieved almost 1000 nodes

2004: SBNS achieves 100,000 nodes

Average BDeu scores. ($s = 4$, $mfss = 4$; 250,000 random edges considered for hillclimbing)

dataset	rand hlclmb	*SBNS*	*SBNS*+MIe	*SBNS*+MIe+2nd	*SBNS*+MIe+2nd+hlclmb
citeseer	-33.26	-27.466	-27.375	-27.273	-26.962
imdb	-121.00	-113.15	-112.45	-112.18	-111.28
institute	-11.87	-13.28	-13.18	-13.13	-12.08
drinks	-6.72	-7.21	-7.02	-7.01	-6.705

Number of links in the resulting nets. ($s = 4$, $mfss = 4$; 100,000 random edges considered for hillclimbing)

dataset	rand hlclmb	*SBNS*	*SBNS*+MIe	*SBNS*+MIe+2nd	*SBNS*+MIe+2nd+hlclmb
citeseer	88,259	29,004	48,724	53,790	116,558
imdb	112,773	33,434	52,376	57,236	111,281
institute	1,672	346	398	457	1,159
drinks	723	51	123	133	709

Timing (min)

dataset	rand hillclmb	*SBNS*	*SBNS*+MIe
citeseer	171	59.8	87
imdb	193	225.6	252.8
institute	.53	.016	.017
drinks	.37	.016	.017

Inference
(Link Completion: Who else?)

from citeseer

Given: few people from Daphne Koller's group:

d_koller, a_pfeffer, l_getoor, b_taskar

Query: Which 1 other person is most likely?

Answer: n_friedman (score -22.524)
s_tong (score -22.695)

Query: Which 2 other people are most likely?

Answer: {a_grove,j_halpern} (score -24.066)
{s_russell, r_parr} (score -24.689)

Conclusions

- Tractable solution to Bayes Net structural learning for massive sparse datasets
- Empirically: obtain less complex networks while maintaining higher accuracy than random hillclimbing
- New prospective on what to do with Frequent Sets ☺

Outline

Cached Sufficient Statistics

Ball Trees Refresher

K-nearest-neighbor classification (exploiting the question part one)

Non-parametric classification

Biosurveillance and Epidemiology

Scan Statistics (exploiting the question part two)

Bayesian Network Learning

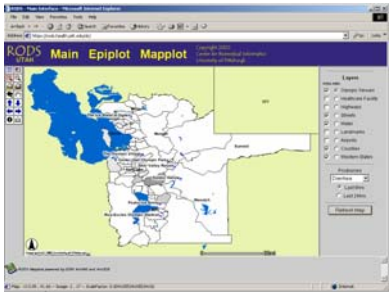
Finding Higher Order Correlations with Frequent Sets(exploiting the question part three)

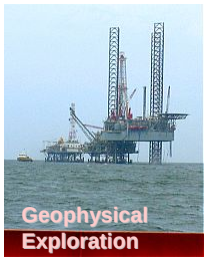
➤ Sensible conclusion

Flaky conclusion

Where are these used?

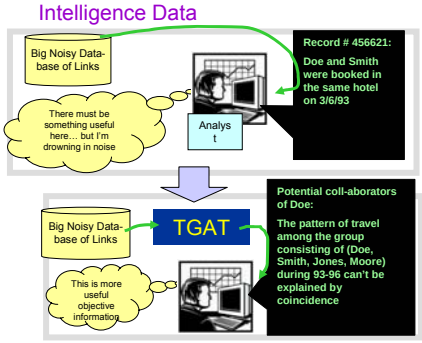
Biomedical Security (with Mike Wagner, University of Pittsburgh)





Autonomous self-tweaking engines

Intelligence Data



Justifiable Conclusions

Justifiable Conclusions

- Geometry can help tractability of Massive Statistical Data Analysis
- Cached sufficient statistics are one approach
- Not merely for simple friendly aggregates

Papers, tutorials, software, data, examples:

www.autonlab.org

Justifiable Conclusions

- Geometry can help tractability of Massive Statistical Data Analysis
- Cached sufficient statistics are one approach
- Not merely for simple friendly aggregates

Fluffy Conclusion

"Theorem of Statistical Computation Benevolence"

If Statistics thinks you're going the right way, it will throw in computational opportunities for you

Papers, tutorials, software, data, examples:

www.autonlab.org