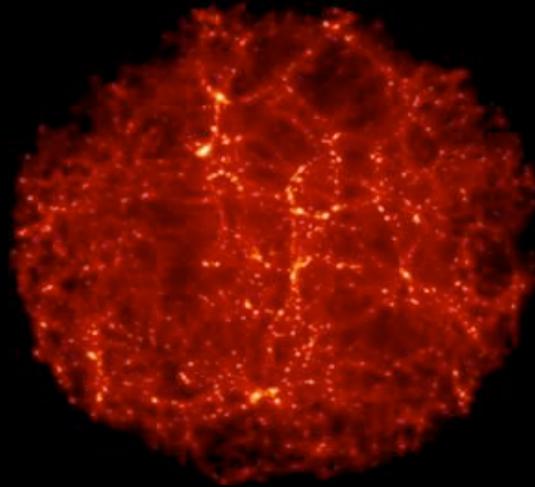


# How to do Machine Learning on Massive Astronomical Datasets



**Alexander Gray**

**Georgia Institute of Technology**  
Computational Science and Engineering  
College of Computing

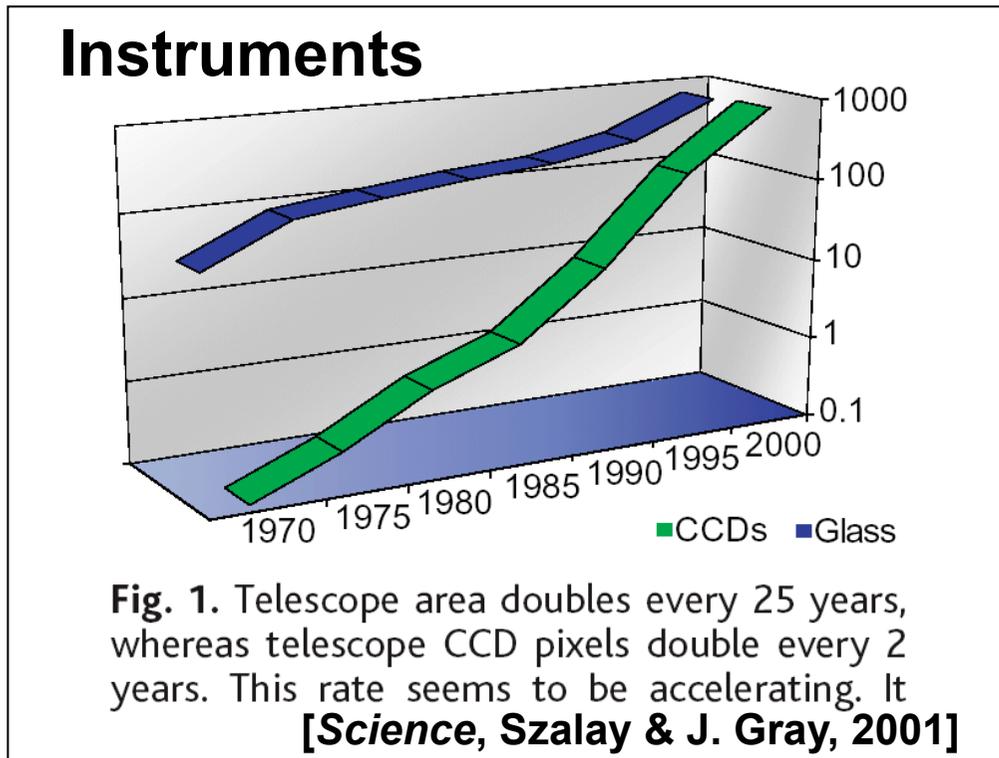
**FASTlab:** Fundamental Algorithmic and Statistical Tools Laboratory

# The FASTlab

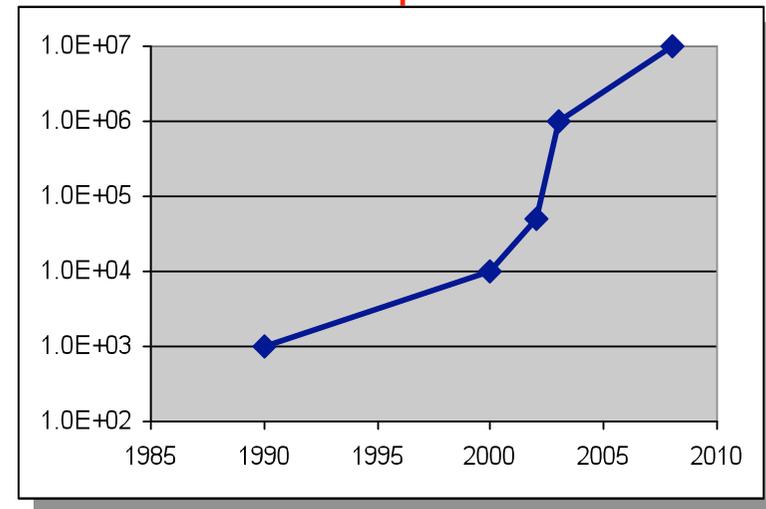
## Fundamental Algorithmic and Statistical Tools Laboratory

1. Arkadas Ozakin: **Research scientist**, PhD Theoretical Physics
2. Dong Ryeol Lee: **PhD student**, CS + Math
3. Ryan Riegel: **PhD student**, CS + Math
4. Parikshit Ram: **PhD student**, CS + Math
5. William March: **PhD student**, Math + CS
6. James Waters: **PhD student**, Physics + CS
7. Hua Ouyang: **PhD student**, CS
8. Sooraj Bhat: **PhD student**, CS
9. Ravi Sastry: **PhD student**, CS
10. Long Tran: **PhD student**, CS
11. Michael Holmes: **PhD student**, CS + Physics (co-supervised)
12. Nikolaos Vasiloglou: **PhD student**, EE (co-supervised)
13. Wei Guan: **PhD student**, CS (co-supervised)
14. Nishant Mehta: **PhD student**, CS (co-supervised)
15. Wee Chin Wong: **PhD student**, ChemE (co-supervised)
16. Abhimanyu Aditya: **MS student**, CS
17. Yatin Kanetkar: **MS student**, CS
18. Praveen Krishnaiah: **MS student**, CS
19. Devika Karnik: **MS student**, CS
20. Prasad Jakka: **MS student**, CS

# Exponential growth in dataset sizes



## Data: CMB Maps



**1990 COBE**                    **1,000**  
**2000 Boomerang** **10,000**  
**2002 CBI**  
   **50,000**  
**2003 WMAP**            **1 Million**  
**2008 Planck**        **10 Million**

## Data: Local Redshift Surveys

**1986 CfA**        **3,500**  
**1996 LCRS**    **23,000**  
**2003 2dF**    **250,000**  
**2005 SDSS** **800,000**

## Data: Angular Surveys

**1970 Lick**        **1M**  
**1990 APM**        **2M**  
**2005 SDSS** **200M**  
**2010 LSST**        **2B**

1993-1999: DPOSS

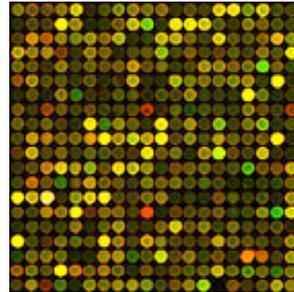
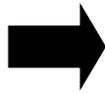
1999-2008: SDSS

Coming: Pan-STARRS, LSST



# Happening everywhere!

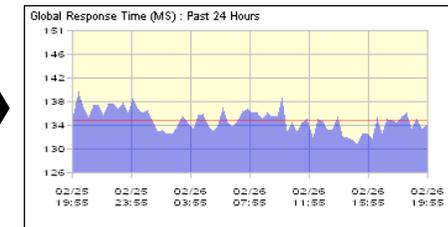
microarray chips      **Molecular biology  
(cancer)**



fiber optics



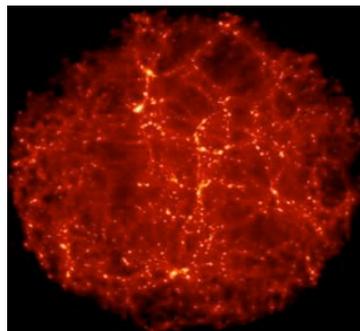
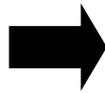
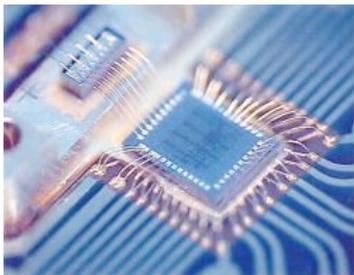
**Network traffic (spam)**



300M/day

microprocessors

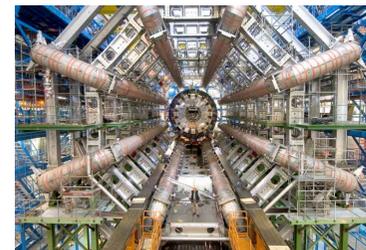
**Simulations  
(Millennium)**



1B

particle colliders

**Particle events (LHC)**



1M/sec

Astrophysicist



Robert Nichol

1. How did galaxies evolve?
2. What was the early universe like?
3. Does dark energy exist?
4. Is our model (GR+inflation) right?

- R. Nichol, Inst. Cosmol. Gravitation
- A. Connolly, U. Pitt Physics
- C. Miller, NOAO
- R. Brunner, NCSA
- G. Djorgovsky, Caltech
- G. Kulkarni, Inst. Cosmol. Gravitation
- D. Wake, Inst. Cosmol. Gravitation
- R. Scranton, U. Pitt Physics
- M. Balogh, U. Waterloo Physics
- I. Szapudi, U. Hawaii Inst. Astronomy
- G. Richards, Princeton Physics
- A. Szalay, Johns Hopkins Physics



Machine learning/  
statistics guy

Astrophysicist



Robert Nichol

1. How did galaxies evolve?
2. What was the early universe like?
3. Does dark energy exist?
4. Is our model (GR+inflation) right?

- Kernel density estimator
- n-point spatial statistics
- Nonparametric Bayes classifier
- Support vector machine
- Nearest-neighbor statistics
- Gaussian process regression
- Hierarchical clustering



Machine learning/  
statistics guy

- R. Nichol, Inst.
- A. Connolly, U
- C. Miller, NOAC
- R. Brunner, NO
- G. Djorgovsky
- G. Kulkarni, In
- D. Wake, Inst. C
- R. Scranton, U. Pitt Physics
- M. Balogh, U. Waterloo Physics
- I. Szapudi, U. Hawaii Inst. Astro.
- G. Richards, Princeton Physics
- A. Szalay, Johns Hopkins Physics

Astrophysicist



Robert Nichol

1. How did galaxies evolve?
2. What was the early universe like?
3. Does dark energy exist?
4. Is our model (GR+inflation) right?

- Kernel density estimator  $O(N^2)$
- n-point spatial statistics  $O(N^n)$
- Nonparametric Bayes classifier  $O(N^2)$
- Support vector machine  $O(N^3)$
- Nearest-neighbor statistics  $O(N^2)$
- Gaussian process regression  $O(N^3)$
- Hierarchical clustering  $O(N^3)$



Machine learning/  
statistics guy

- R. Nichol, Inst.
- A. Connolly, U
- C. Miller, NOAC
- R. Brunner, NC
- G. Djorgovsky
- G. Kulkarni, In
- D. Wake, Inst. C
- R. Scranton, U. Pitt Physics
- M. Balogh, U. Waterloo Physics
- I. Szapudi, U. Hawaii Inst. Astro.
- G. Richards, Princeton Physics
- A. Szalay, Johns Hopkins Physics



Carnegie Mellon as

Astrophysicist



Robert Nichol

1. How did galaxies evolve?
2. What was the early universe like?
3. Does dark energy exist?
4. Is our model (GR+inflation) right?

- Kernel density estimator  $O(N^2)$
- n-point spatial statistics  $O(N^n)$
- Nonparametric Bayes classifier  $O(N^2)$
- Support vector machine  $O(N^3)$
- Nearest-neighbor statistics  $O(N^2)$
- Gaussian process regression  $O(N^3)$
- Hierarchical clustering  $O(N^3)$



Machine learning/  
statistics guy

- R. Nichol, Inst.
- A. Connolly, U
- C. Mil... NOAC
- R. Bru... r, NC
- G. Dj... sky
- G. Kul...
- D. Wal...
- R. Scra... Physics
- M. Del...

**But I have 1 million points**

- G. Richardson, Princeton Physics
- A. Szalay, Johns Hopkins Physics



Carnegie Mellon as

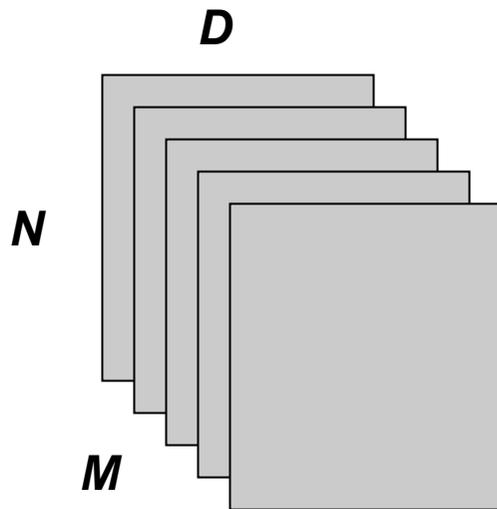
# The challenge

## State-of-the-art statistical methods...

- *Best accuracy with fewest assumptions*

**with orders-of-mag more efficiency.**

- **Large  $N$**  (#data),  **$D$**  (#features),  **$M$**  (#models)



Reduce data? Use simpler model?

Approximation with poor/no error bounds?

→ **Poor results**

# How to do Machine Learning on Massive Astronomical Datasets?

1. Choose the appropriate **statistical task and method** for the scientific question
2. Use the fastest **algorithm and data structure** for the statistical method
3. Put it in **software**

# How to do Machine Learning on Massive Astronomical Datasets?

1. Choose the appropriate **statistical task and method** for the scientific question
2. Use the fastest **algorithm and data structure** for the statistical method
3. Put it in **software**

# 10 data analysis problems, and scalable tools we'd like for them

1. **Querying** (*e.g. characterizing a region of space*):
  - spherical range-search  $O(N)$
  - orthogonal range-search  $O(N)$
  - k-nearest-neighbors  $O(N)$
  - all-k-nearest-neighbors  $O(N^2)$
2. **Density estimation** (*e.g. comparing galaxy types*):
  - mixture of Gaussians
  - kernel density estimation  $O(N^2)$
  - $L_2$  density tree [Ram and Gray in prep]
  - manifold kernel density estimation  $O(N^3)$  [Ozakin and Gray 2008, to be submitted]
  - hyper-kernel density estimation  $O(N^4)$  [Sastry and Gray 2008, submitted]

# 10 data analysis problems, and scalable tools we'd like for them

## 3. Regression (*e.g. photometric redshifts*):

- linear regression  $O(D^2)$
- kernel regression  $O(N^2)$
- Gaussian process regression/kriging  $O(N^3)$

## 4. Classification (*e.g. quasar detection, star-galaxy separation*):

- k-nearest-neighbor classifier  $O(N^2)$
- nonparametric Bayes classifier  $O(N^2)$
- support vector machine (SVM)  $O(N^3)$
- *non-negative SVM  $O(N^3)$  [Guan and Gray, in prep]*
- *false-positive-limiting SVM  $O(N^3)$  [Sastry and Gray, in prep]*
- *separation map  $O(N^3)$  [Vasiloglou, Gray, and Anderson 2008, submitted]*

# 10 data analysis problems, and scalable tools we'd like for them

## 5. Dimension reduction (*e.g. galaxy or spectra characterization*):

- principal component analysis  $O(D^2)$
- non-negative matrix factorization
- kernel PCA  $O(N^3)$
- maximum variance unfolding  $O(N^3)$
- *co-occurrence embedding  $O(N^3)$  [Ozakin and Gray, in prep]*
- *rank-based manifolds  $O(N^3)$  [Ouyang and Gray 2008, ICML]*
- *isometric non-negative matrix factorization  $O(N^3)$  [Vasiloglou, Gray, and Anderson 2008, submitted]*

## 6. Outlier detection (*e.g. new object types, data cleaning*):

- by density estimation, by dimension reduction
- *by robust  $L_p$  estimation [Ram, Riegel and Gray, in prep]*

# 10 data analysis problems, and scalable tools we'd like for them

## 7. Clustering (*e.g. automatic Hubble sequence*)

- by dimension reduction, by density estimation
- k-means
- mean-shift segmentation  $O(N^2)$
- hierarchical clustering (“friends-of-friends”)  $O(N^3)$

## 8. Time series analysis (*e.g. asteroid tracking, variable objects*):

- Kalman filter  $O(D^2)$
- hidden Markov model  $O(D^2)$
- trajectory tracking  $O(N^n)$
- *Markov matrix factorization [Tran, Wong, and Gray 2008, submitted]*
- *functional independent component analysis [Mehta and Gray 2008, submitted]*

# 10 data analysis problems, and scalable tools we'd like for them

## 9. Feature selection and causality (*e.g. which features predict star/galaxy*)

- LASSO regression
- $L_1$  SVM
- Gaussian graphical model inference and structure search
- discrete graphical model inference and structure search
- *0-1 feature-selecting SVM [Guan and Gray, in prep]*
- *$L_1$  Gaussian graphical model inference and structure search [Tran, Lee, Holmes, and Gray, in prep]*

## 10. 2-sample testing and matching (*e.g. cosmological validation, multiple surveys*):

- minimum spanning tree  $O(N^3)$
- **$n$ -point correlation  $O(N^n)$**
- *bipartite matching/Gaussian graphical model inference  $O(N^3)$  [Waters and Gray, in prep]*

# How to do Machine Learning on Massive Astronomical Datasets?

1. Choose the appropriate **statistical task and method** for the scientific question
2. Use the fastest **algorithm and data structure** for the statistical method
3. Put it in **software**

# Core computational problems

What are the basic mathematical operations, or bottleneck subroutines, can we focus on developing fast algorithms for?

# Core computational problems

- Aggregations
- Generalized N-body problems
- Graphical model inference
- Linear algebra
- Optimization

# Core computational problems

Aggregations, GNPs, graphical models, linear algebra, optimization

- **Querying:** nearest-neighbor, sph range-search, ortho range-search, all-nn
- **Density estimation:** kernel density estimation, mixture of Gaussians
- **Regression:** linear regression, kernel regression, Gaussian process regression
- **Classification:** nearest-neighbor classifier, nonparametric Bayes classifier, support vector machine
- **Dimension reduction:** principal component analysis, non-negative matrix factorization, kernel PCA, maximum variance unfolding
- **Outlier detection:** by robust  $L_2$  estimation, by density estimation, by dimension reduction
- **Clustering:** k-means, mean-shift, hierarchical clustering (“friends-of-friends”), by dimension reduction, by density estimation
- **Time series analysis:** Kalman filter, hidden Markov model, trajectory tracking
- **Feature selection and causality:** LASSO regression,  $L_1$  support vector machine, Gaussian graphical models, discrete graphical models
- **2-sample testing and matching:** n-point correlation, bipartite matching

# Aggregations

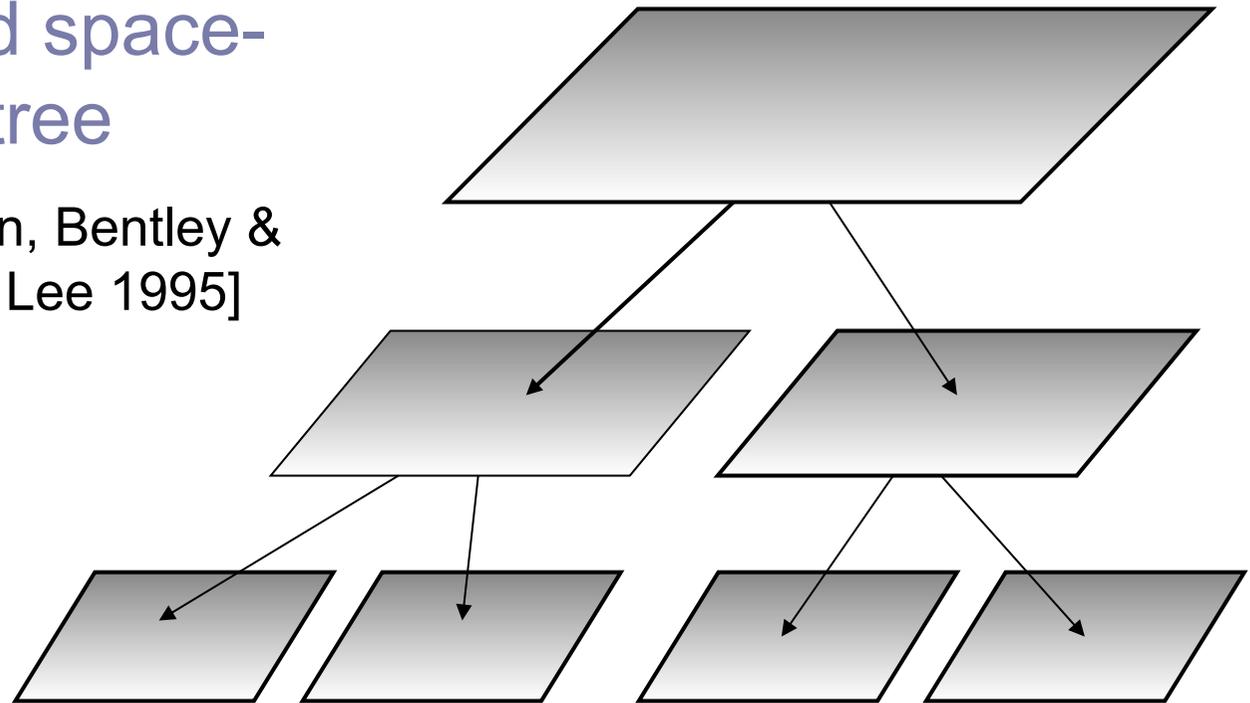
- **How it appears:** nearest-neighbor, sph range-search, ortho range-search
- **Common methods:** locality sensitive hashing, kd-trees, metric trees, disk-based trees
- **Mathematical challenges:** high dimensions, provable runtime, distribution-dependent analysis, parallel indexing
- **Mathematical topics:** computational geometry, randomized algorithms

# How can we compute this efficiently?

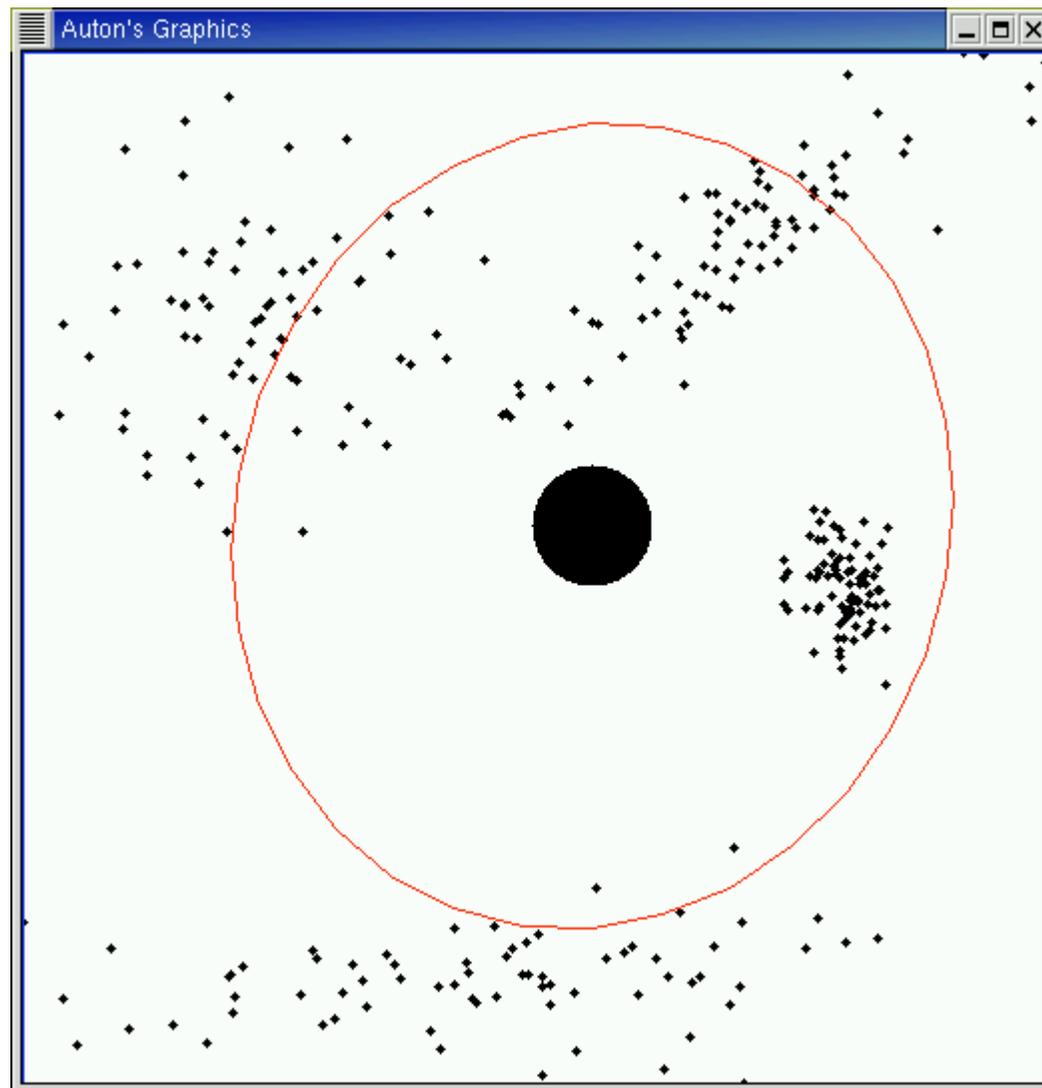
## ***kd-trees:***

most widely-used space-partitioning tree

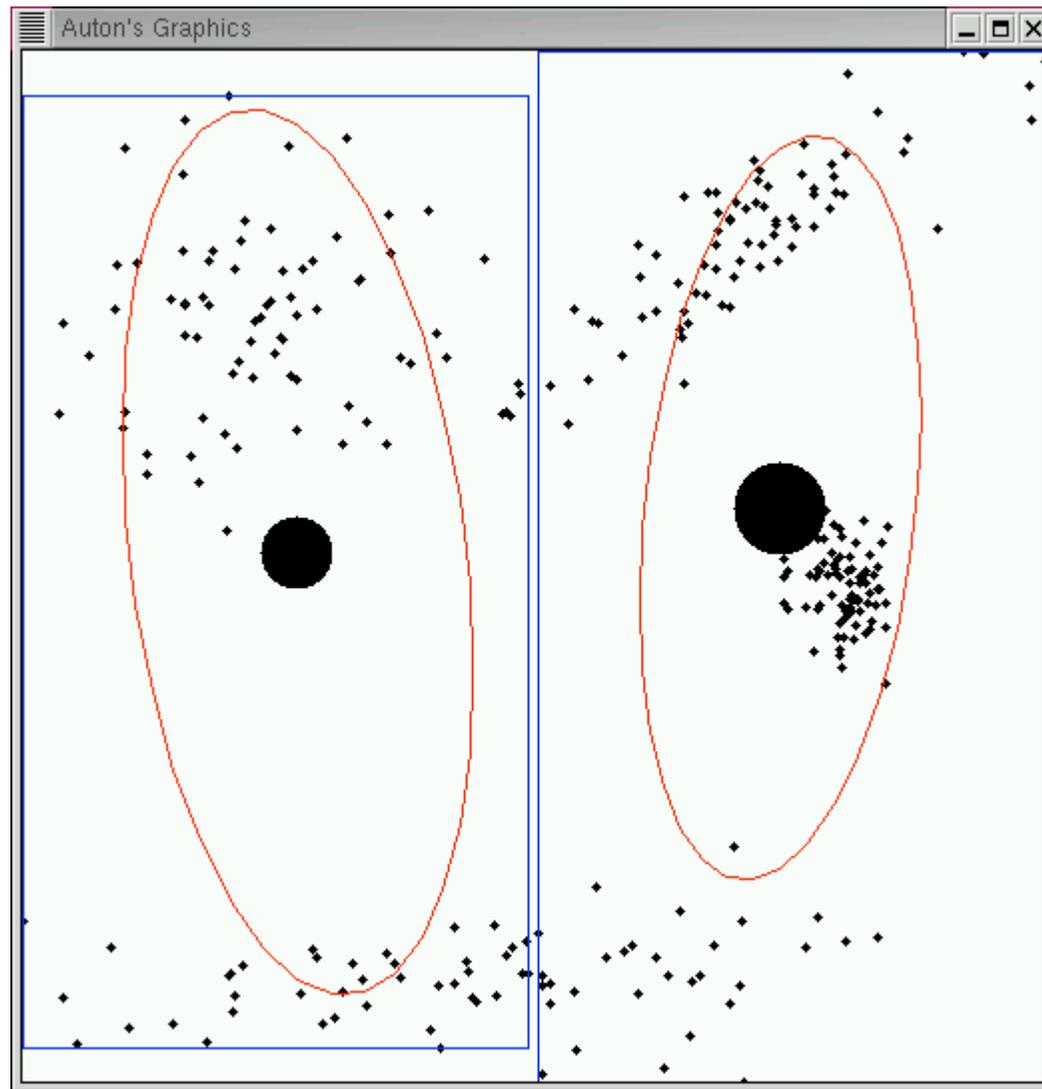
[Bentley 1975], [Friedman, Bentley & Finkel 1977],[Moore & Lee 1995]



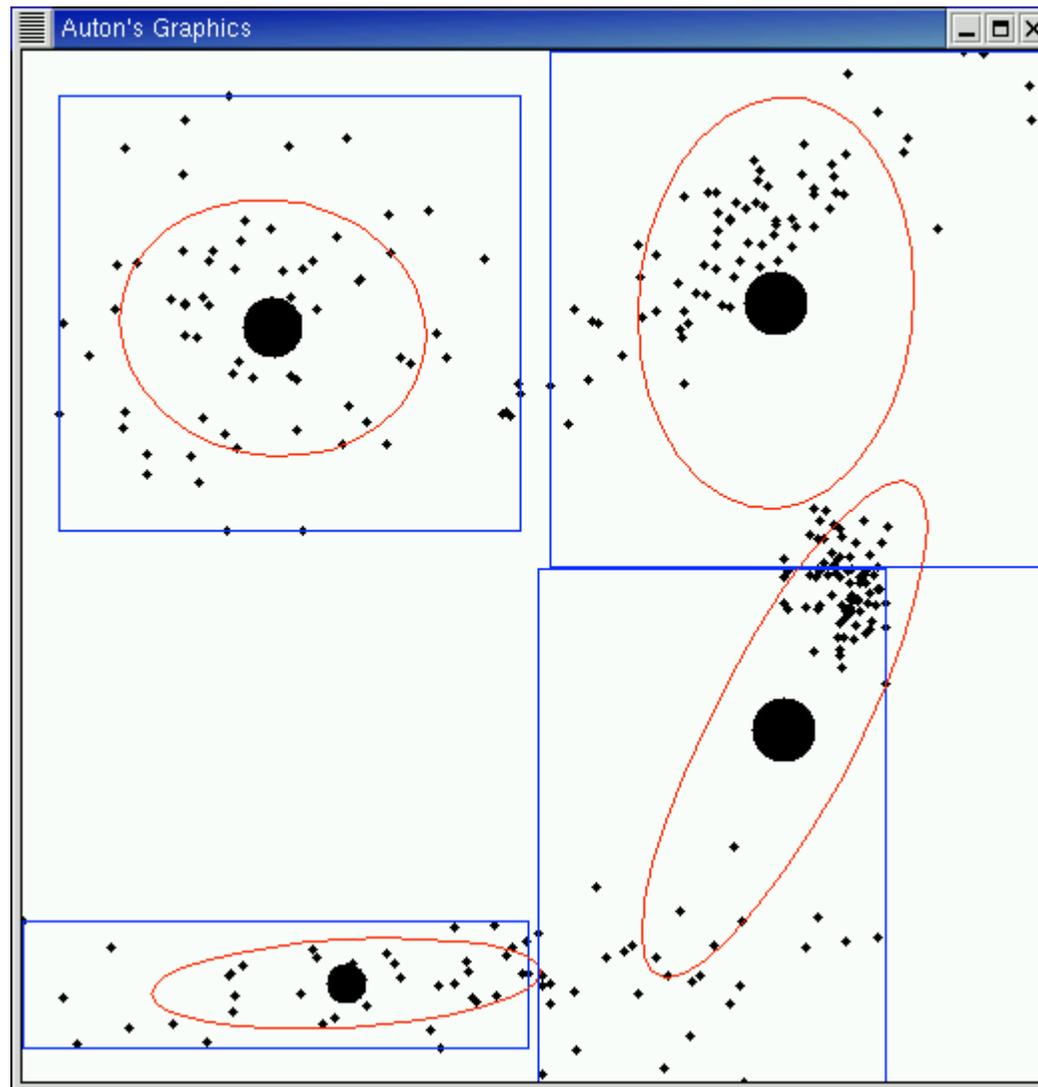
# A *kd*-tree: level 1



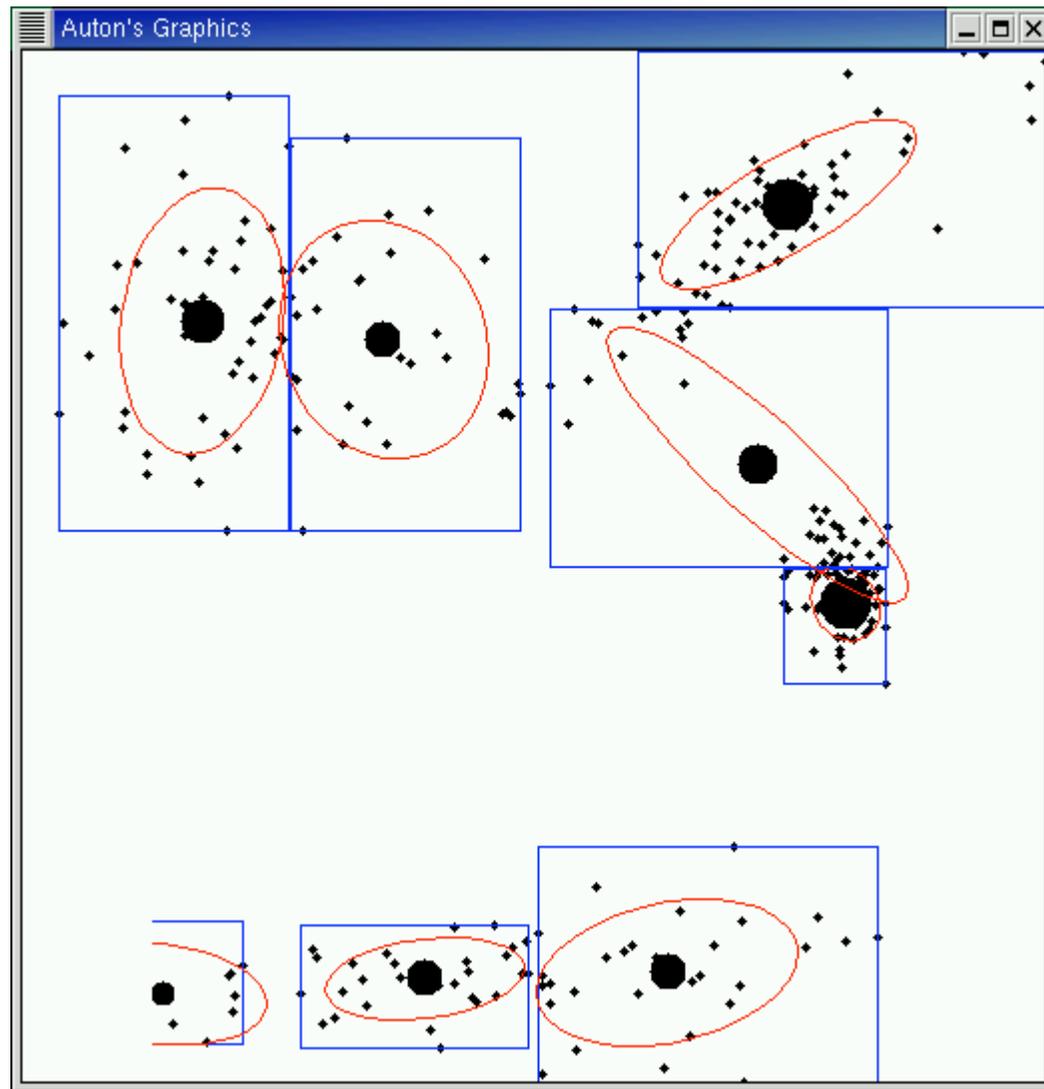
# A *kd*-tree: level 2



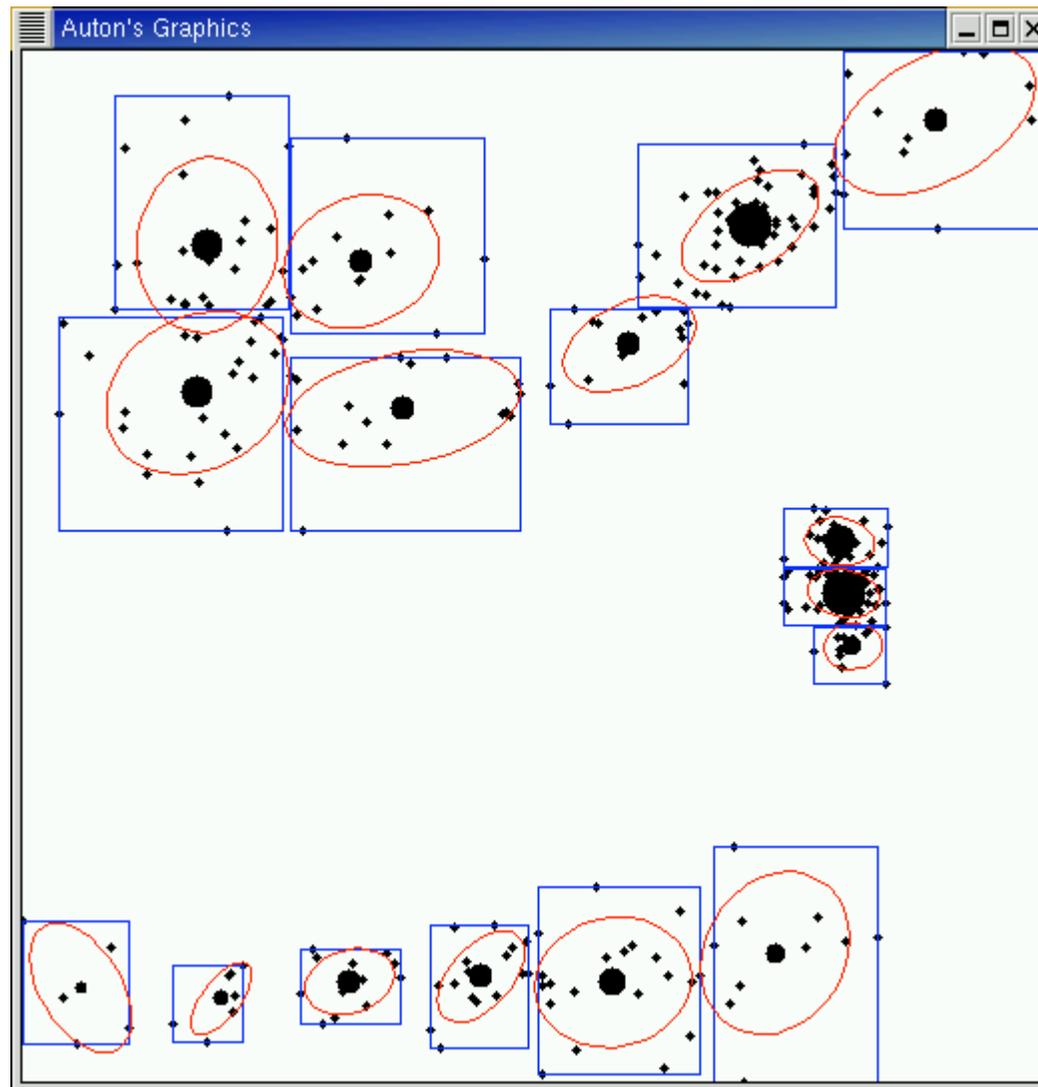
# A *kd*-tree: level 3



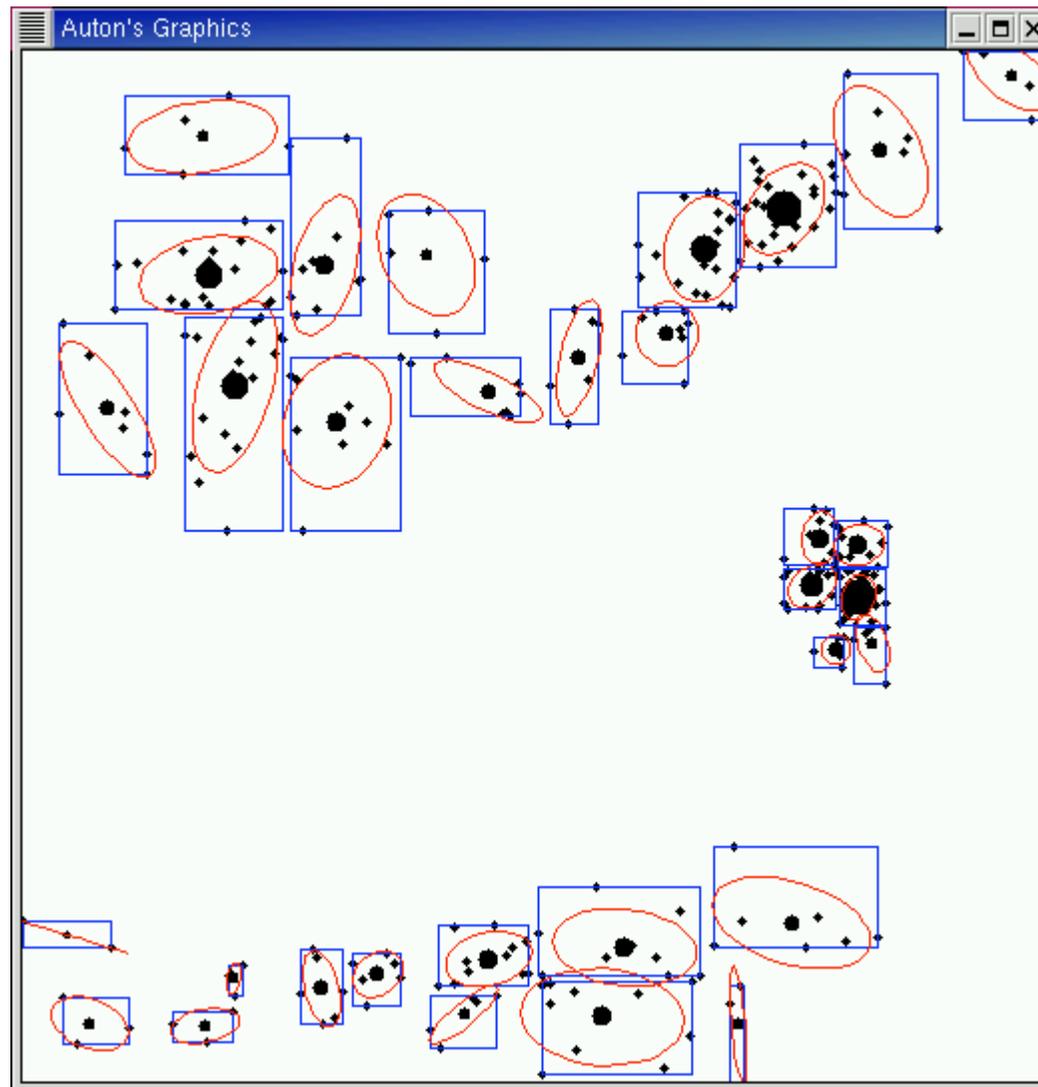
# A *kd*-tree: level 4



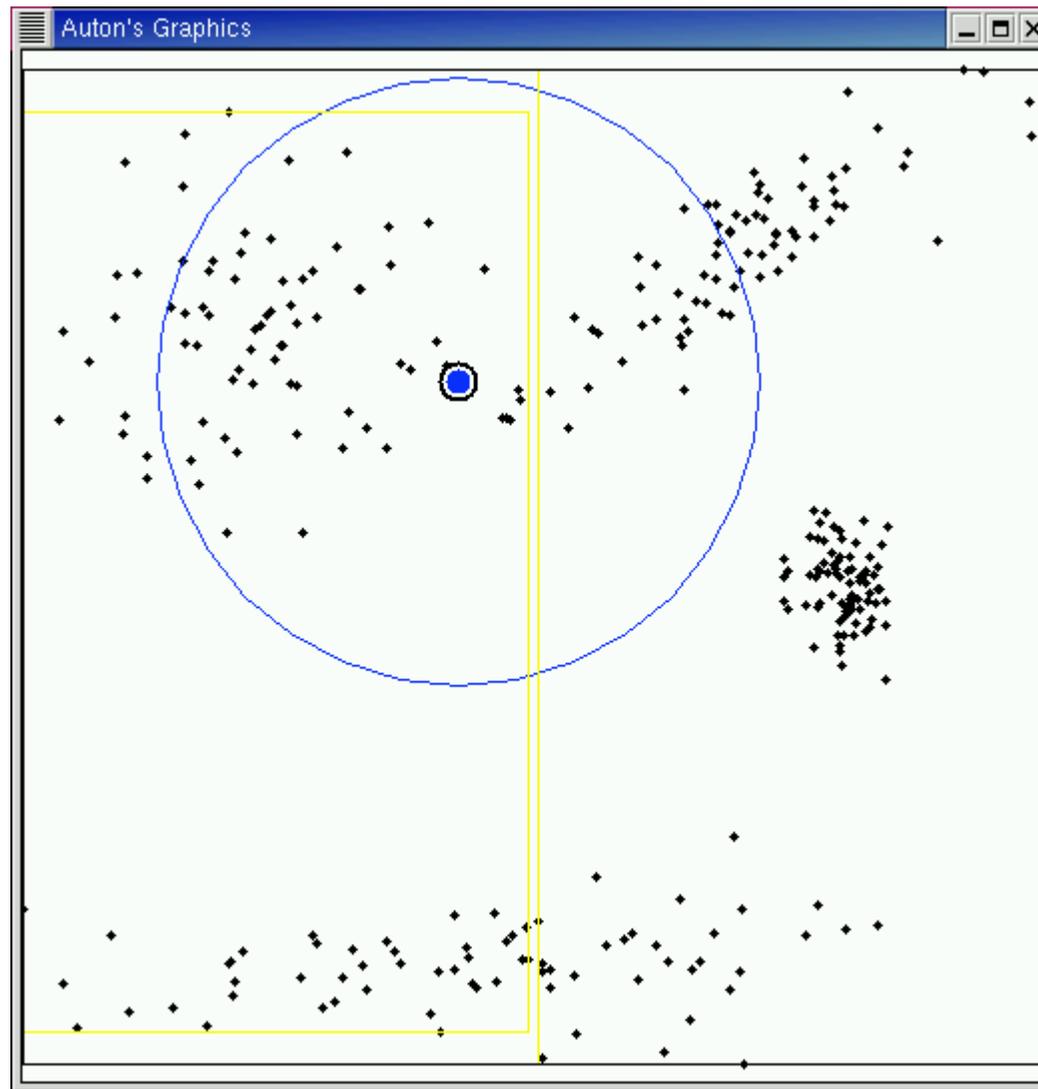
# A *kd*-tree: level 5



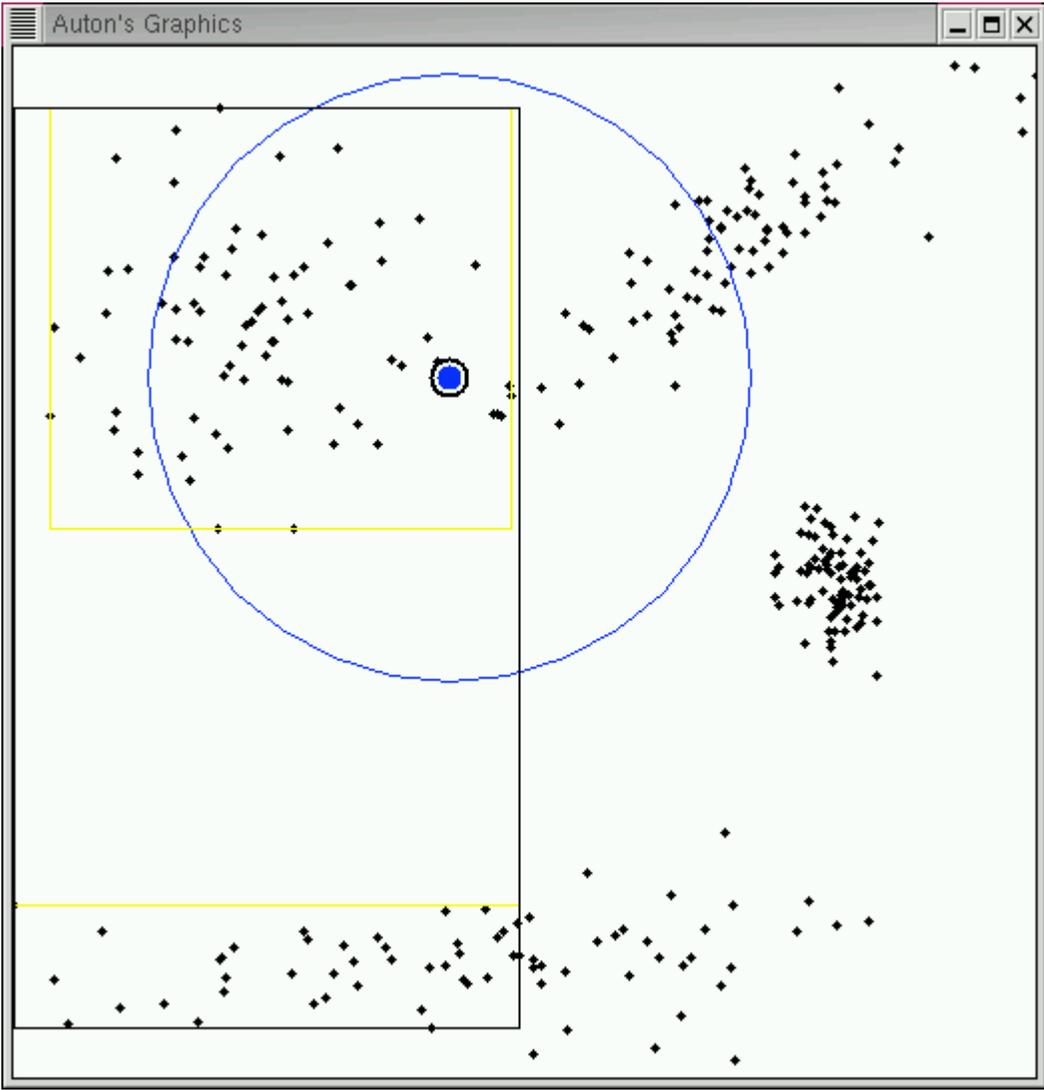
# A *kd*-tree: level 6



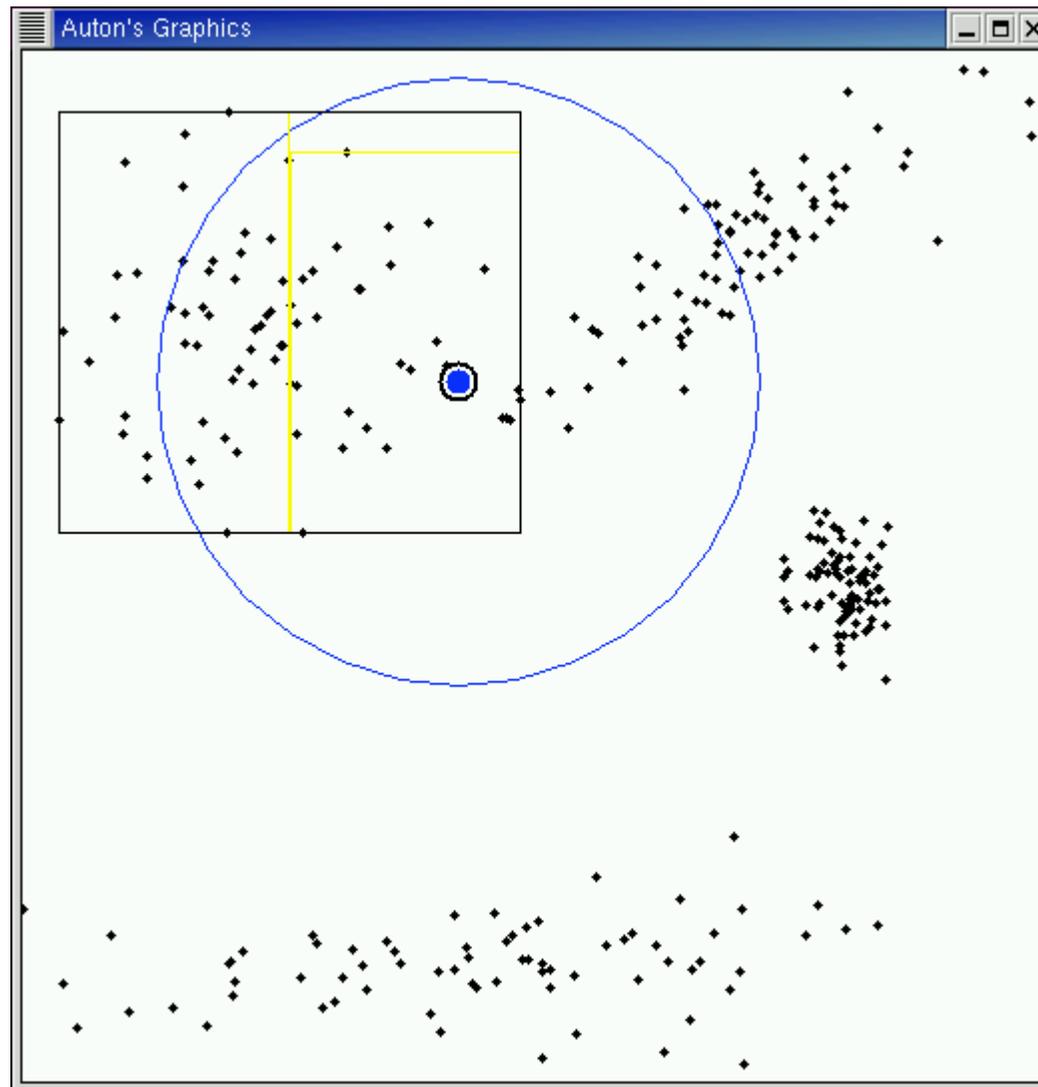
# Range-count recursive algorithm



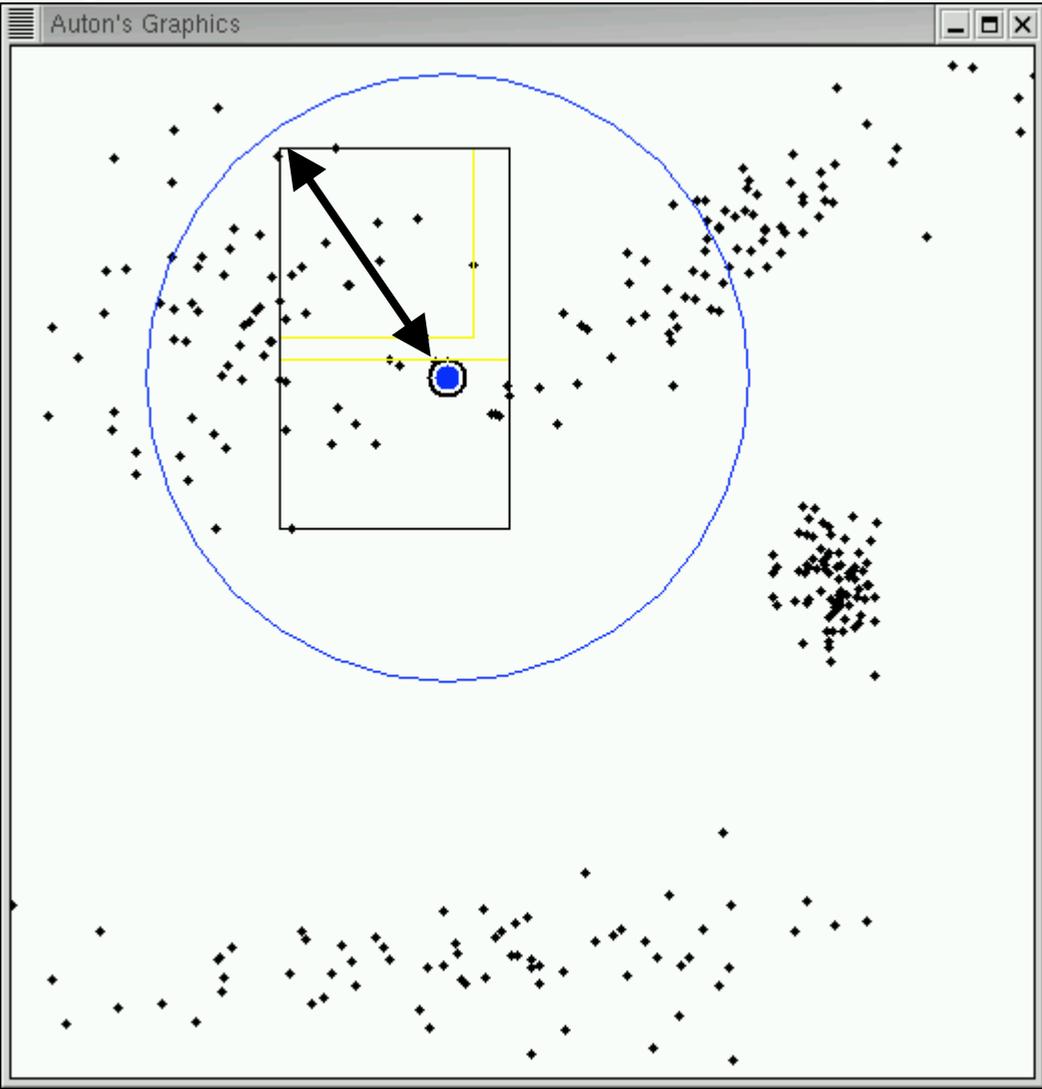
# Range-count recursive algorithm



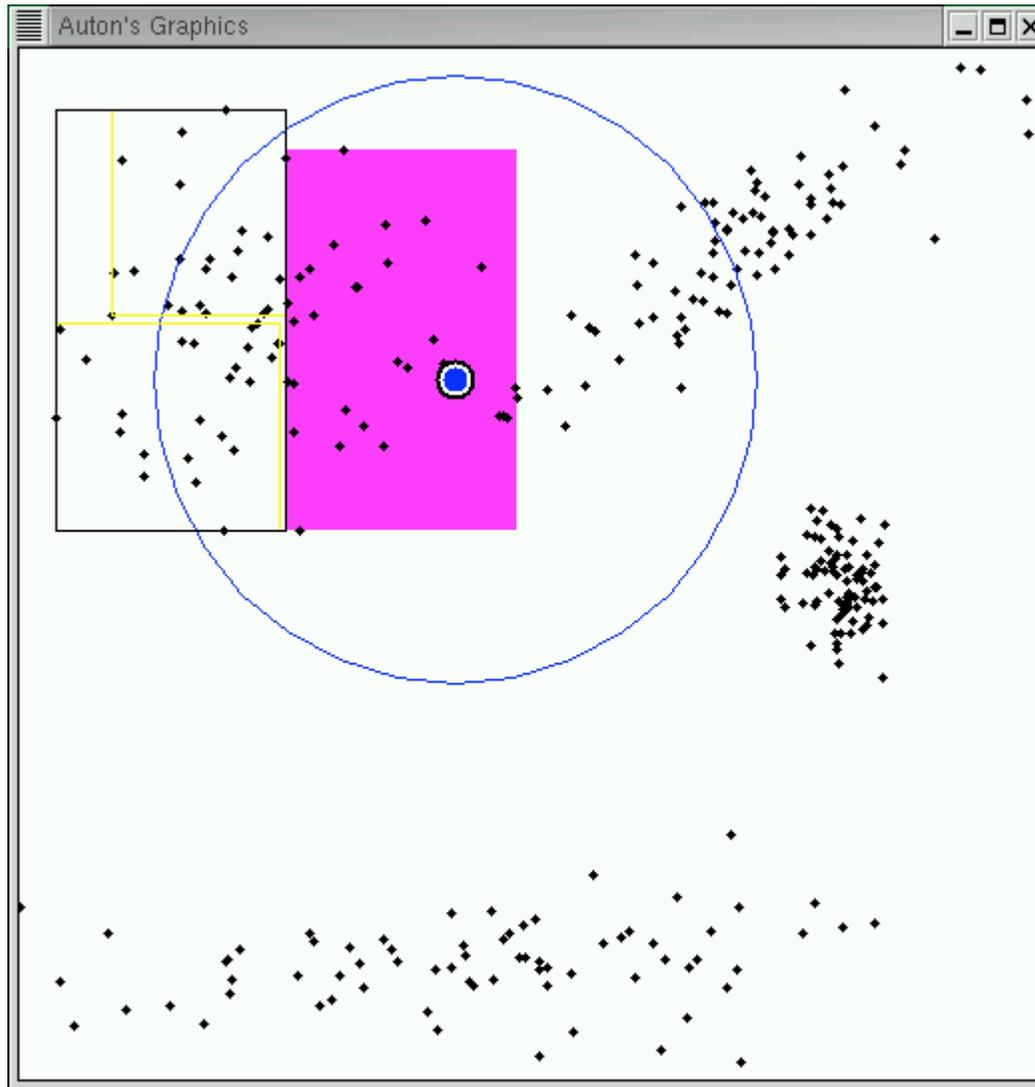
# Range-count recursive algorithm



# Range-count recursive algorithm

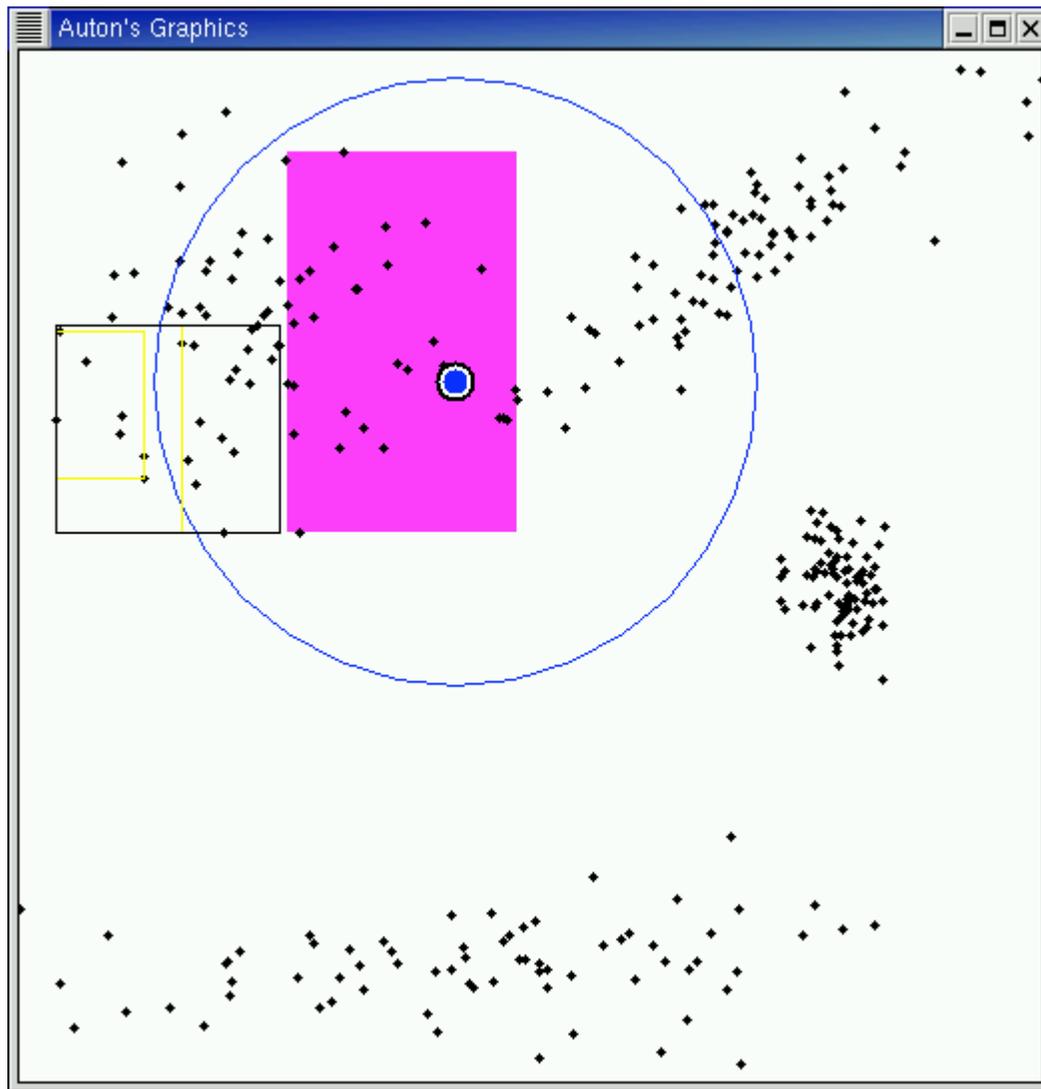


# Range-count recursive algorithm

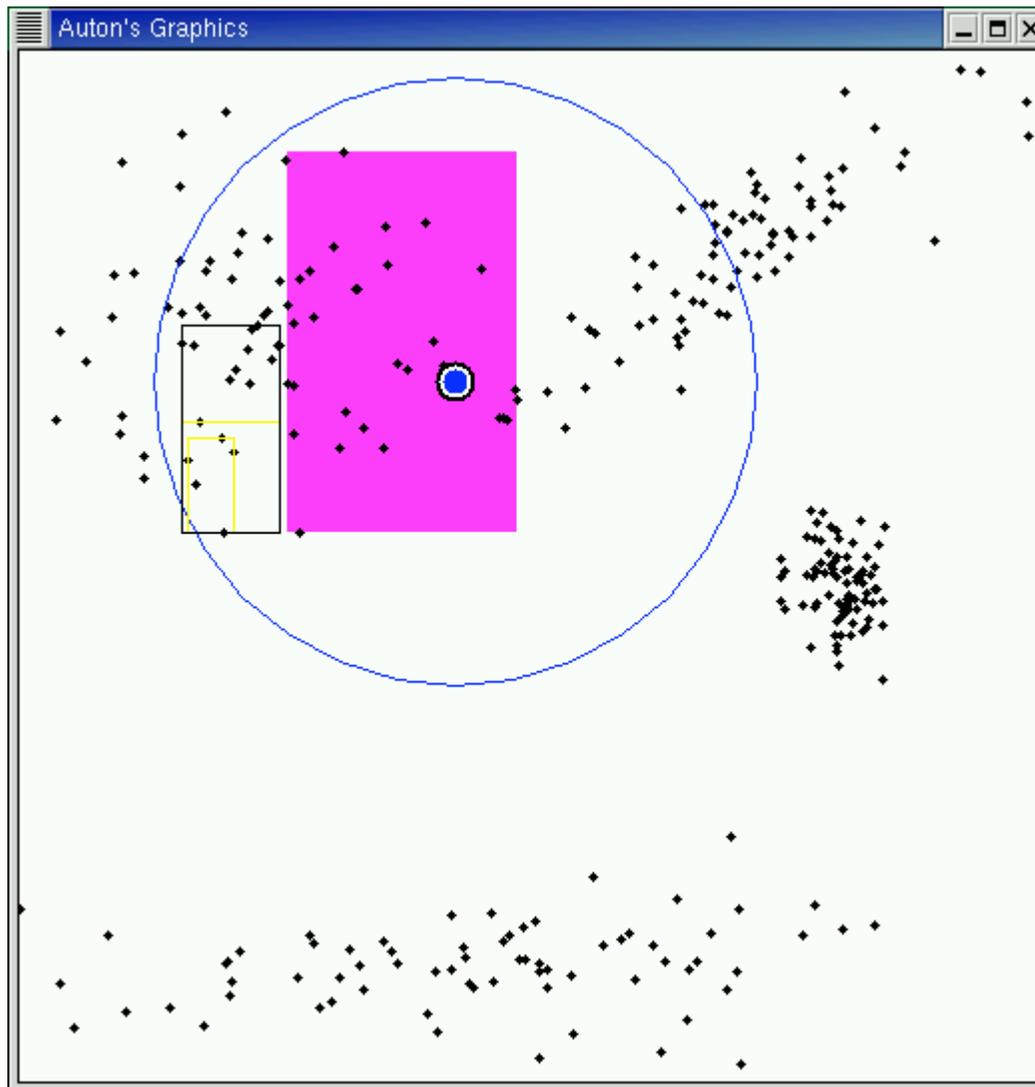


**Pruned!**  
(inclusion)

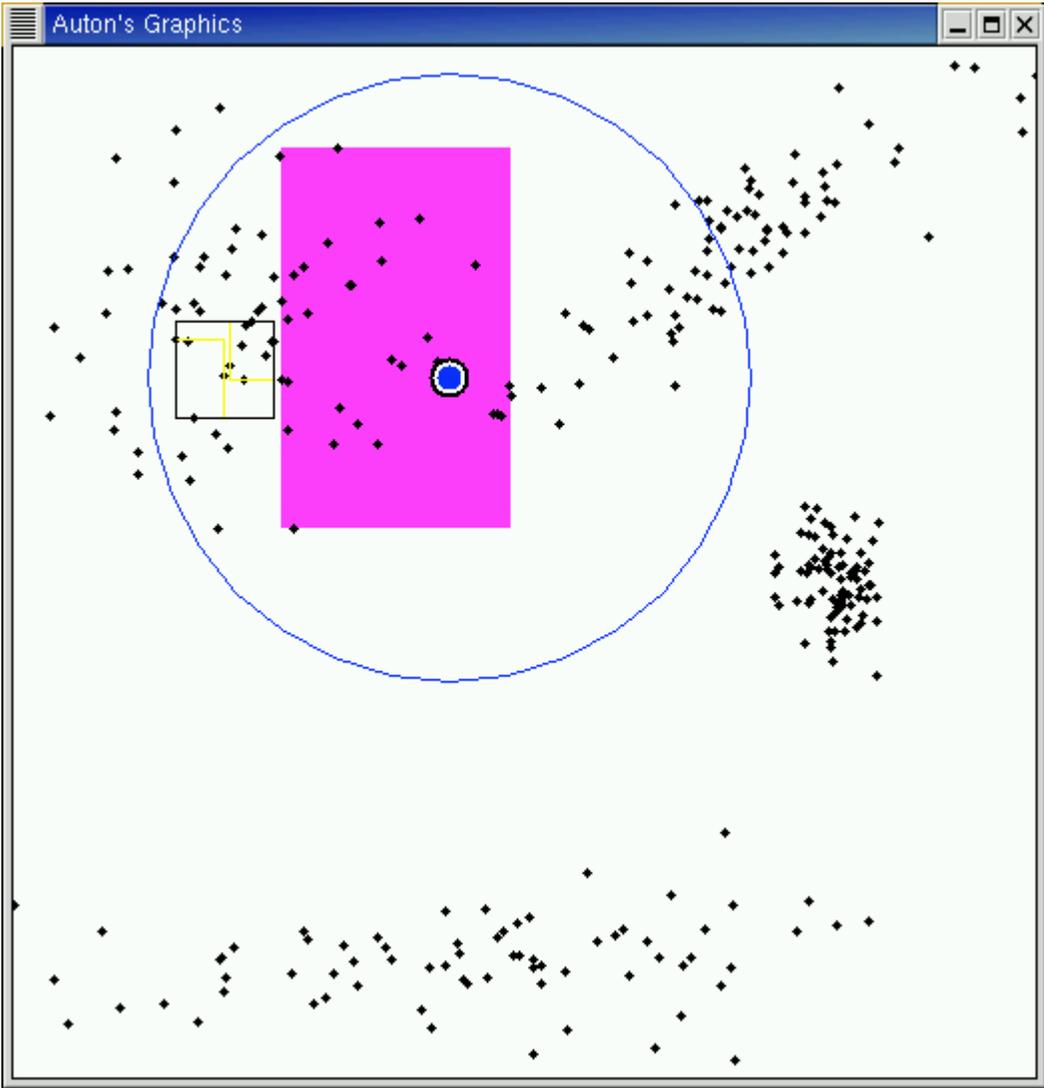
# Range-count recursive algorithm



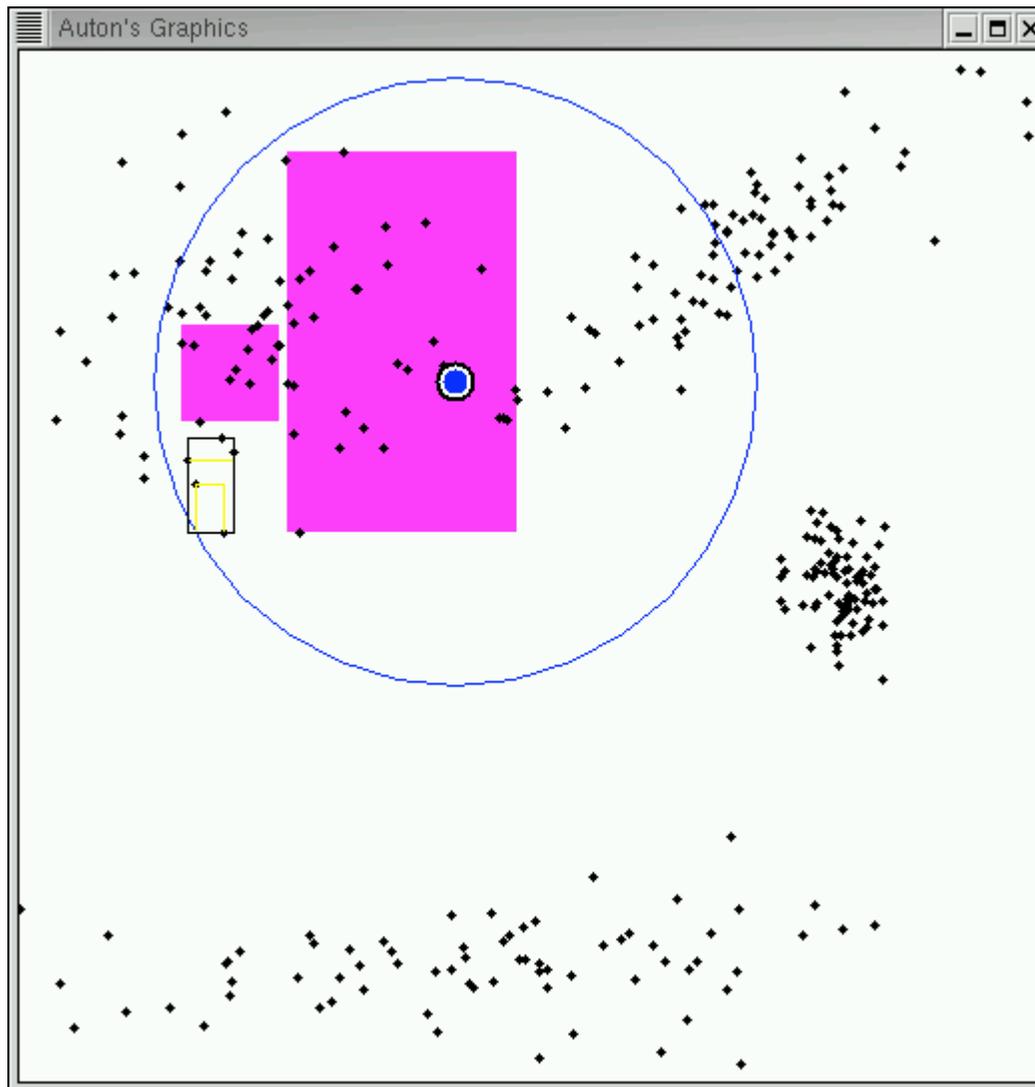
# Range-count recursive algorithm



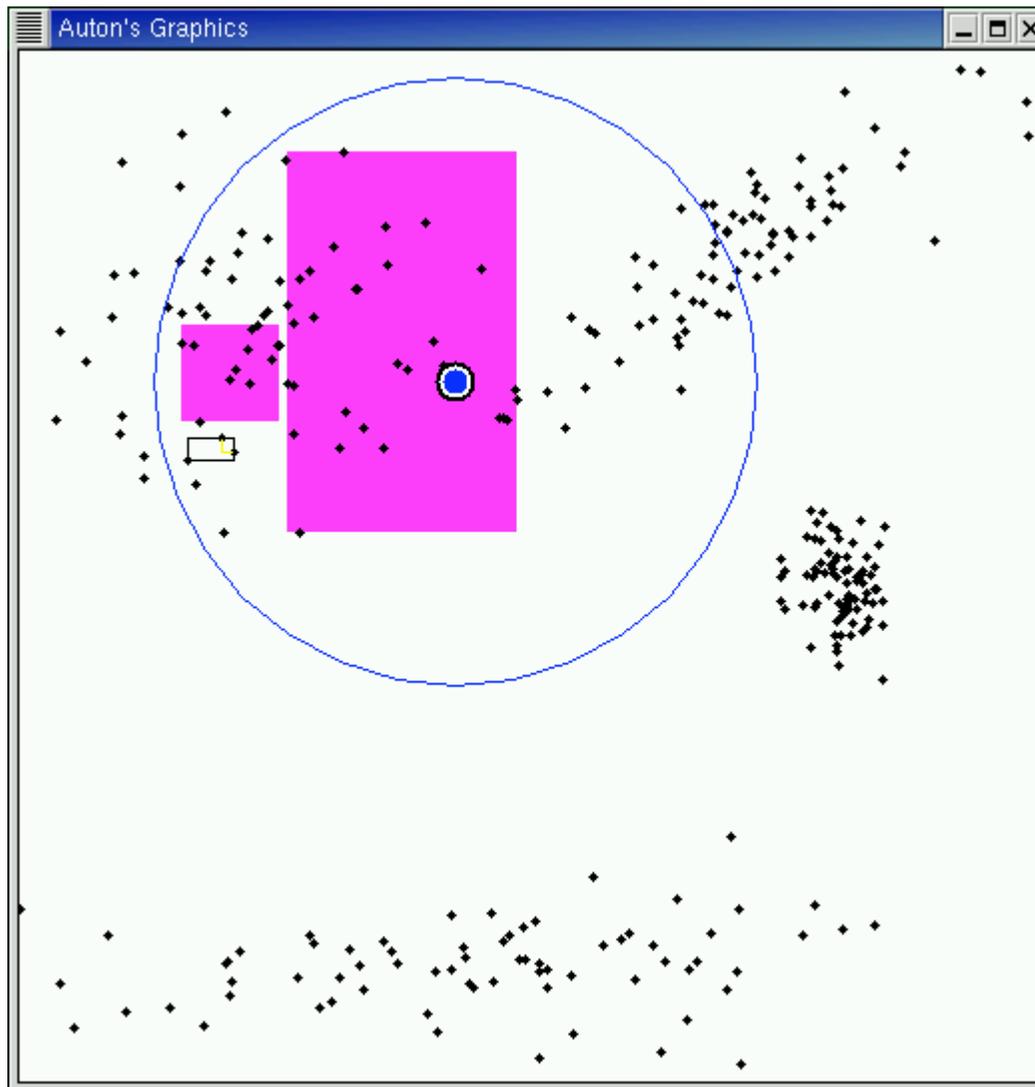
# Range-count recursive algorithm



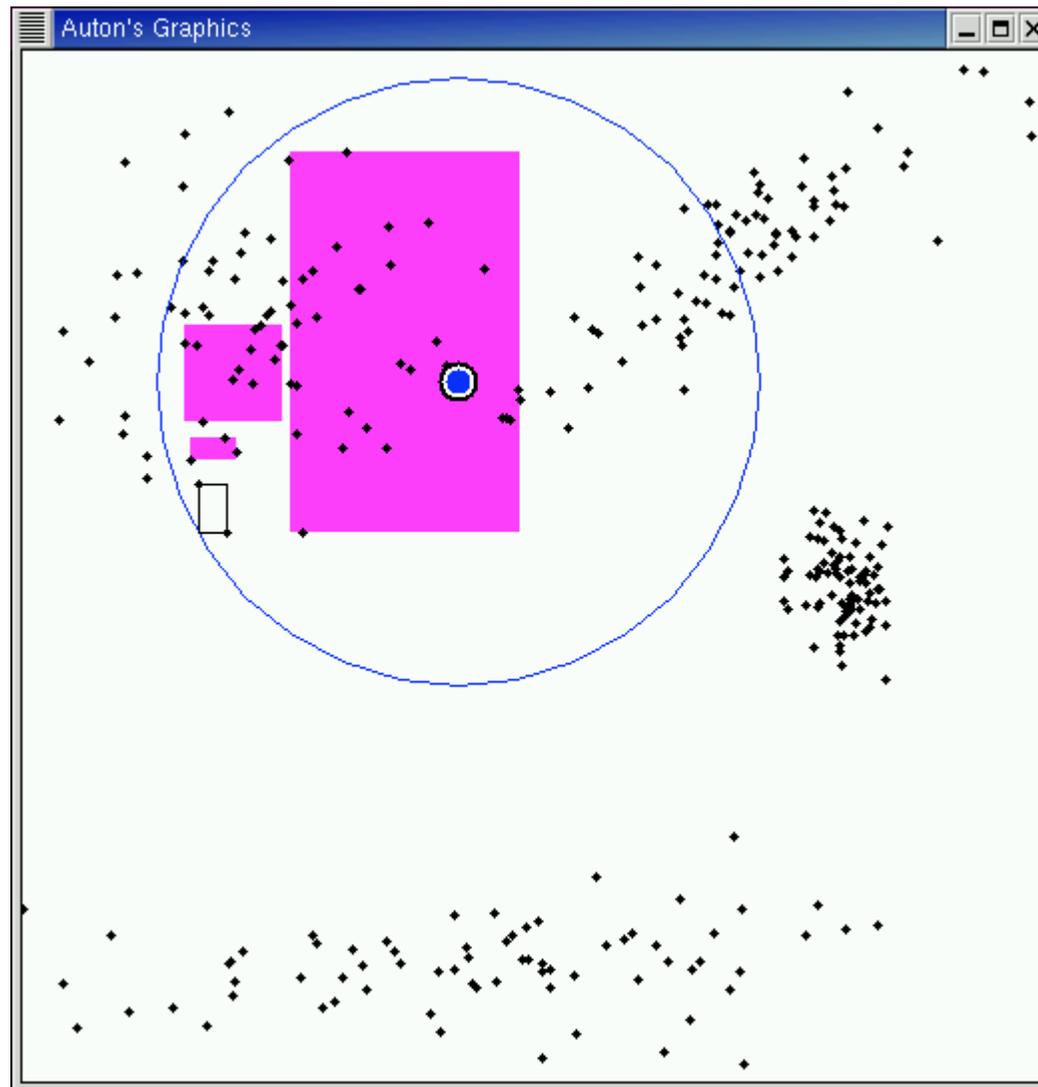
# Range-count recursive algorithm



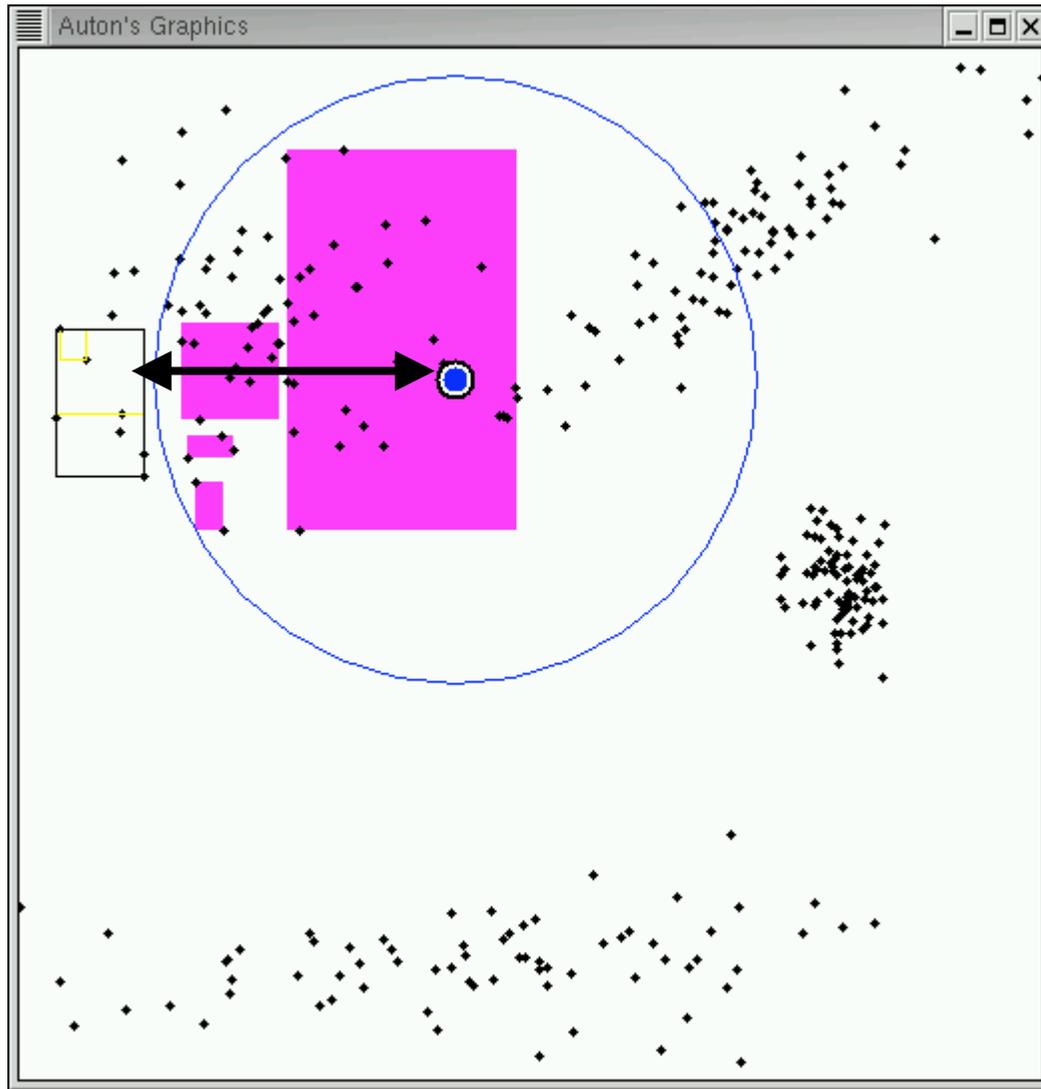
# Range-count recursive algorithm



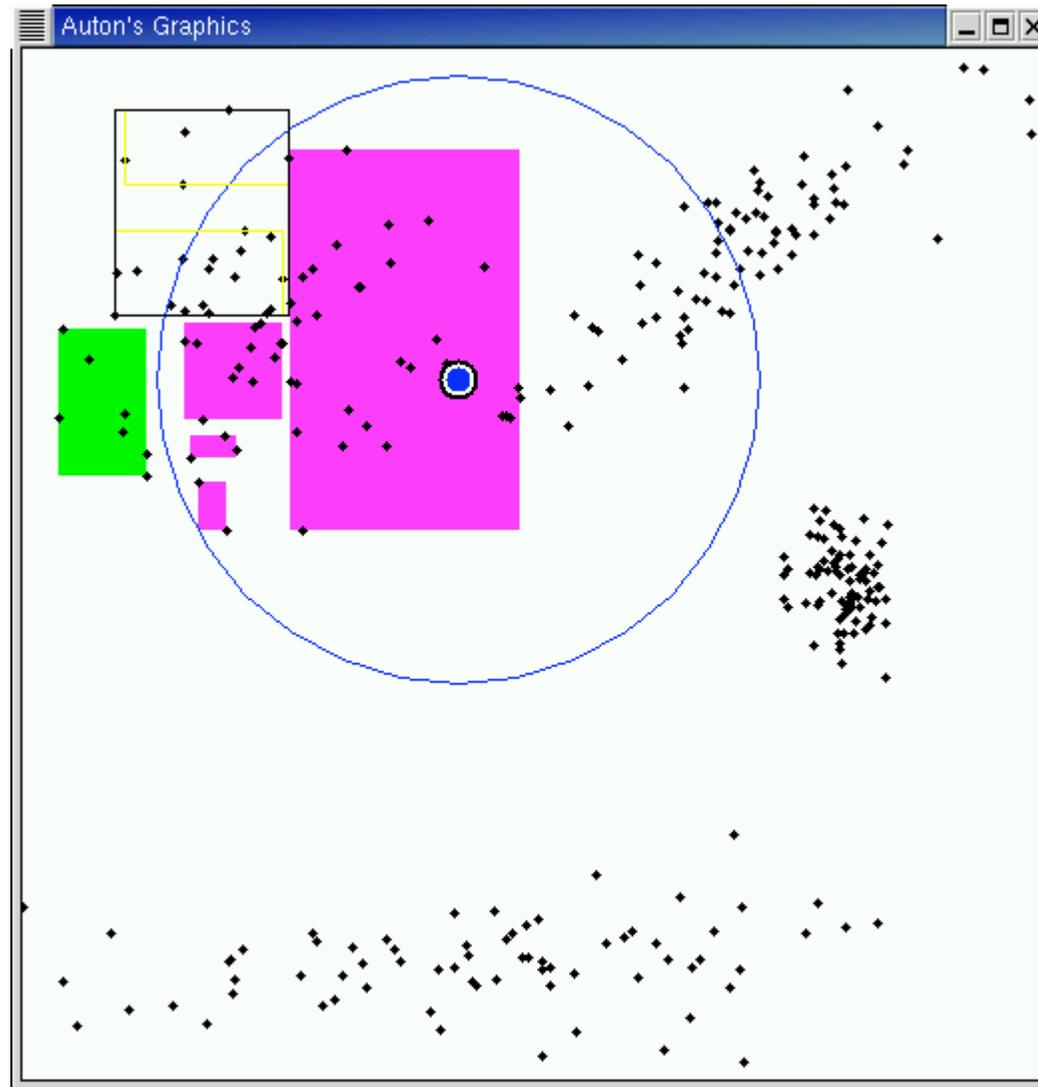
# Range-count recursive algorithm



# Range-count recursive algorithm

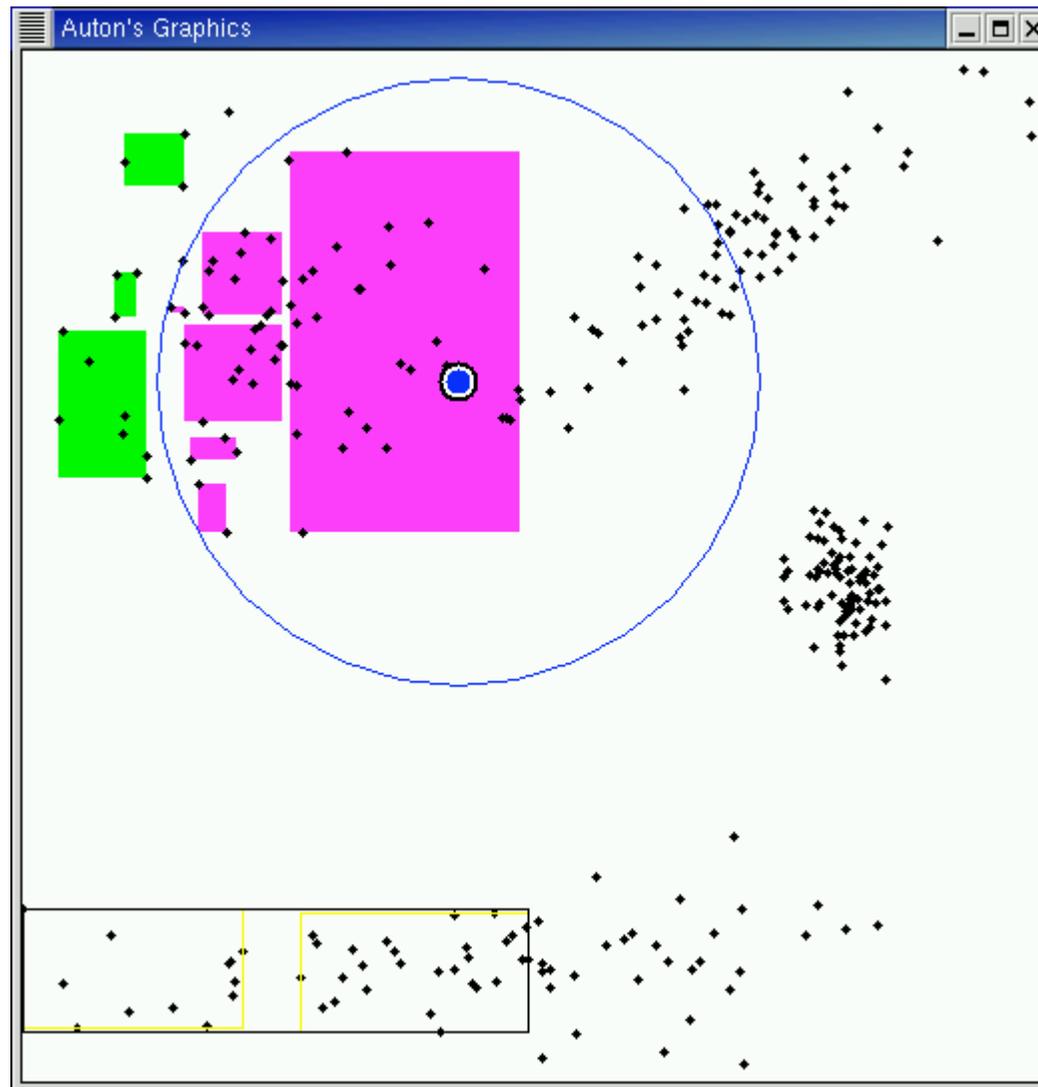


# Range-count recursive algorithm

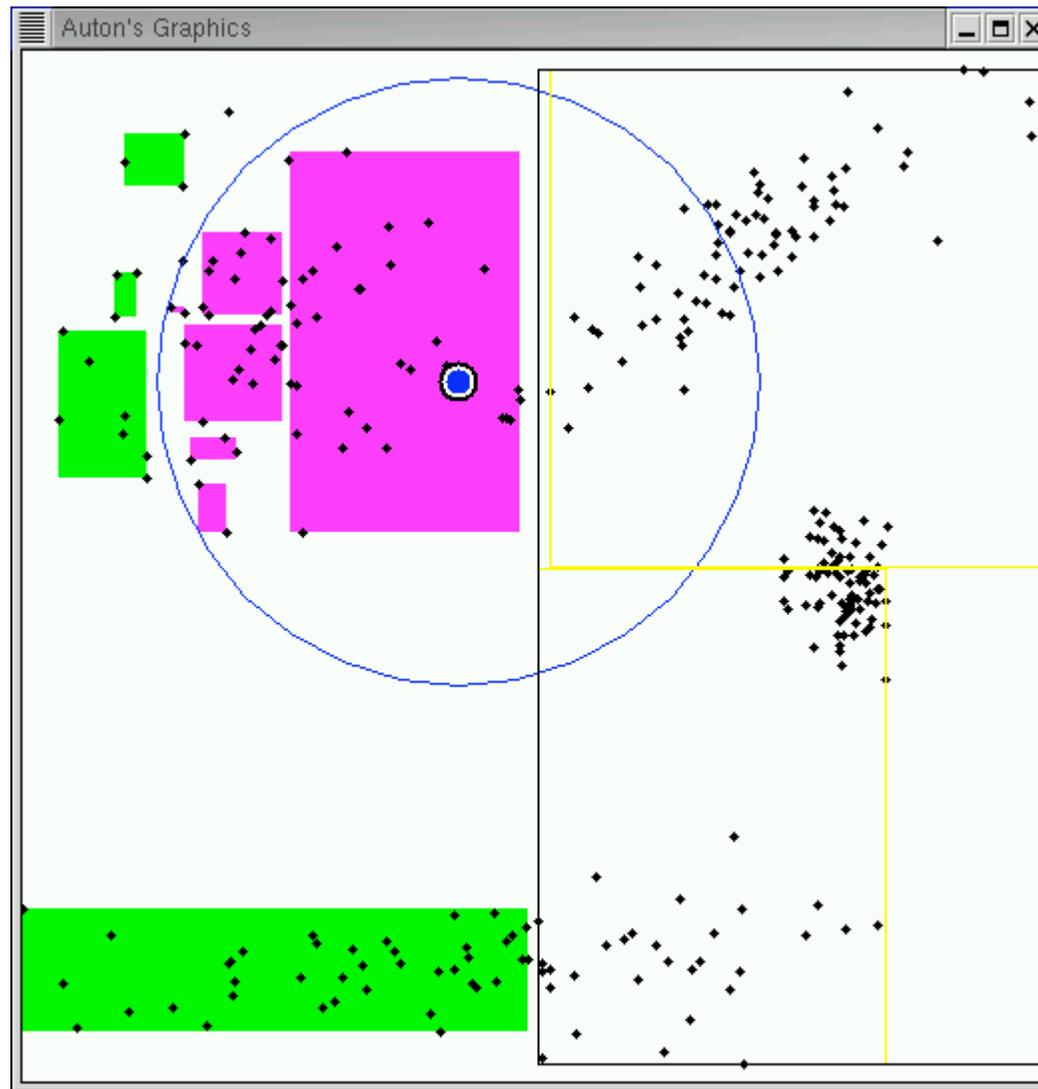


**Pruned!**  
(exclusion)

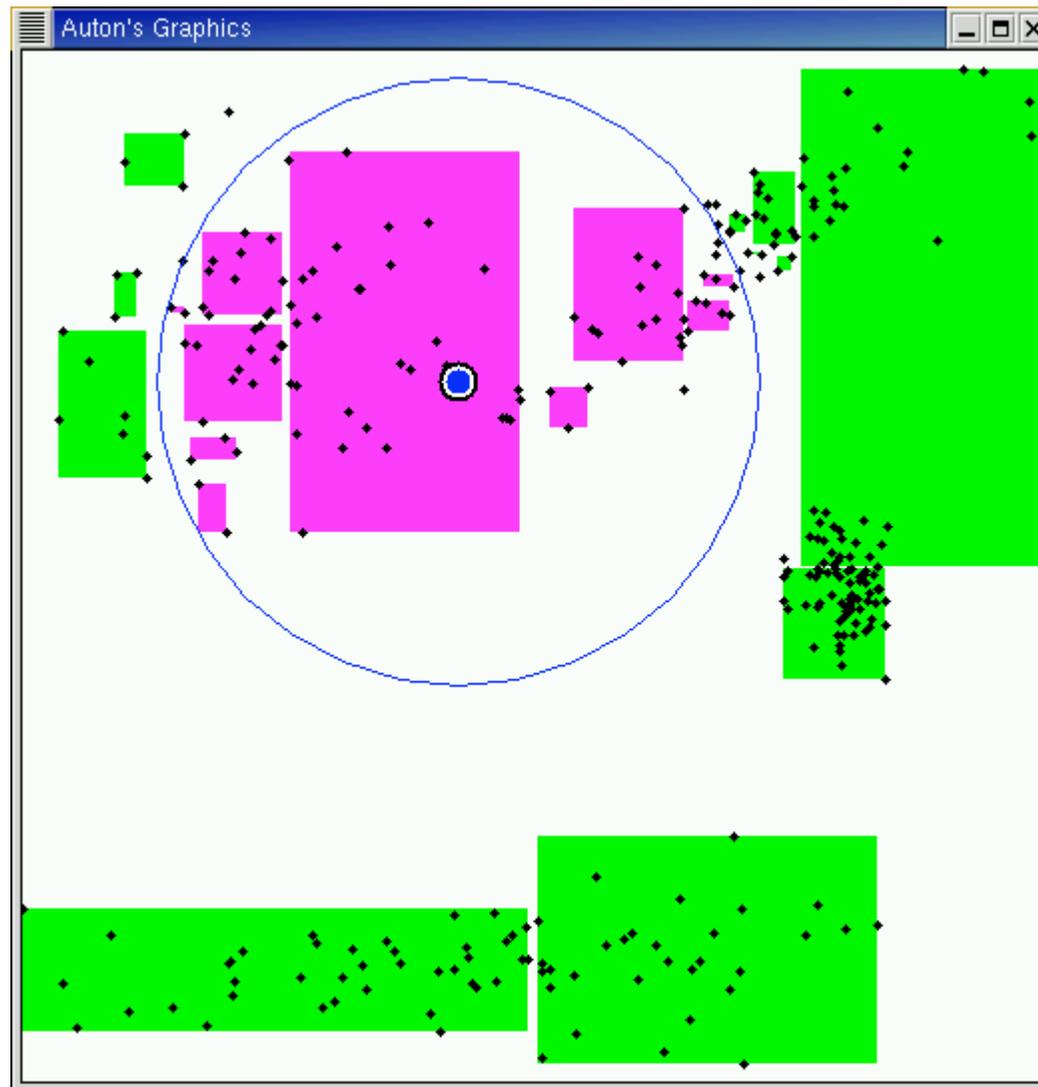
# Range-count recursive algorithm



# Range-count recursive algorithm



# Range-count recursive algorithm



fastest  
practical  
algorithm  
[Bentley 1975]

**our  
algorithms  
can use  
any tree**

# Aggregations

- **Interesting approach:** *Cover-trees [Beygelzimer et al 2004]*
  - Provable runtime
  - Consistently good performance, even in higher dimensions
- **Interesting approach:** *Learning trees [Cayton et al 2007]*
  - Learning data-optimal data structures
  - Improves performance over kd-trees
- **Interesting approach:** *MapReduce [Dean and Ghemawat 2004]*
  - Brute-force
  - But makes HPC automatic for a certain problem form
- **Interesting approach:** *approximation in rank [Ram, Ouyang and Gray]*
  - Approximate NN in terms of distance conflicts with known theoretical results
  - Is approximation in rank feasible?

# Generalized N-body Problems

- **How it appears:** kernel density estimation, mixture of Gaussians, kernel regression, Gaussian process regression, nearest-neighbor classifier, nonparametric Bayes classifier, support vector machine, kernel PCA, hierarchical clustering, trajectory tracking, n-point correlation
- **Common methods:** FFT, Fast Gauss Transform, Well-Separated Pair Decomposition
- **Mathematical challenges:** high dimensions, query-dependent relative error guarantee, parallel, beyond pairwise potentials
- **Mathematical topics:** approximation theory, computational physics, computational geometry

# Generalized N-body Problems

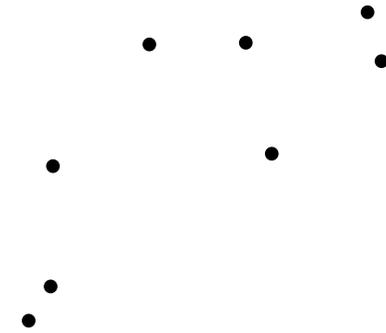
- **Interesting approach:** *Generalized Fast Multipole Method, aka multi-tree methods* [Gray and Moore 2001, NIPS; Riegel, Boyer and Gray]
  - Fastest practical algorithms for the problems to which it has been applied
  - Hard query-dependent relative error bounds
  - Automatic parallelization (*THOR: Tree-based Higher-Order Reduce*) [Boyer, Riegel and Gray to be submitted]

# Characterization of an entire distribution?

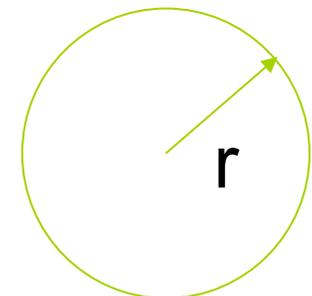
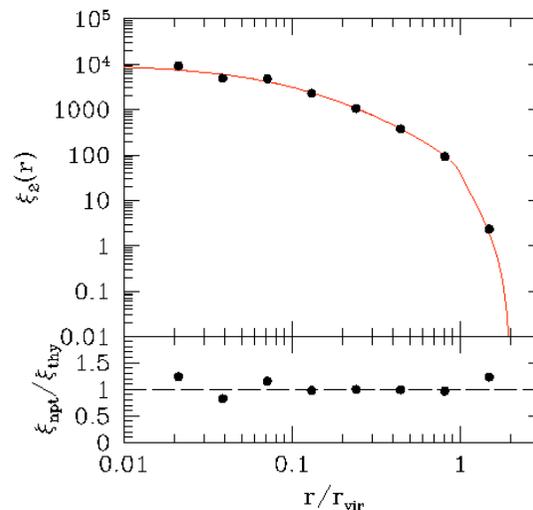
## 2-point correlation

“How many pairs have distance  $< r$  ?”

$$\sum_i^N \sum_{j \neq i}^N I(\|x_i - x_j\| < r)$$



2-point correlation  
function



# The $n$ -point correlation functions

- **Spatial inferences:** filaments, clusters, voids, homogeneity, isotropy, 2-sample testing, ...
- **Foundation** for theory of point processes [Daley, Vere-Jones 1972], unifies spatial statistics [Ripley 1976]
- **Used heavily** in biostatistics, cosmology, particle physics, statistical physics

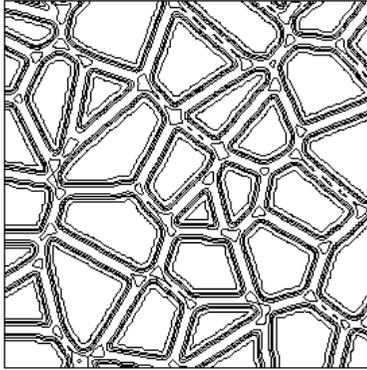
2pcf definition:

$$dP = \lambda^2 dV_1 dV_2 [1 + \xi(r)]$$

3pcf definition:

$$dP = \lambda^3 dV_1 dV_2 dV_3 \cdot [1 + \xi(r_{12}) + \xi(r_{23}) + \xi(r_{13}) + \zeta(r_{12}, r_{23}, r_{13})]$$

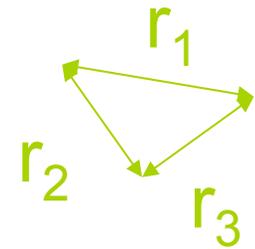
Voronoi foam, smoothed original



Voronoi foam, random phases

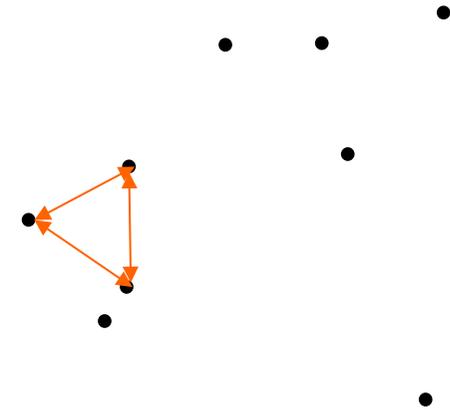


Standard model:  $n > 0$  terms should be zero!



# 3-point correlation

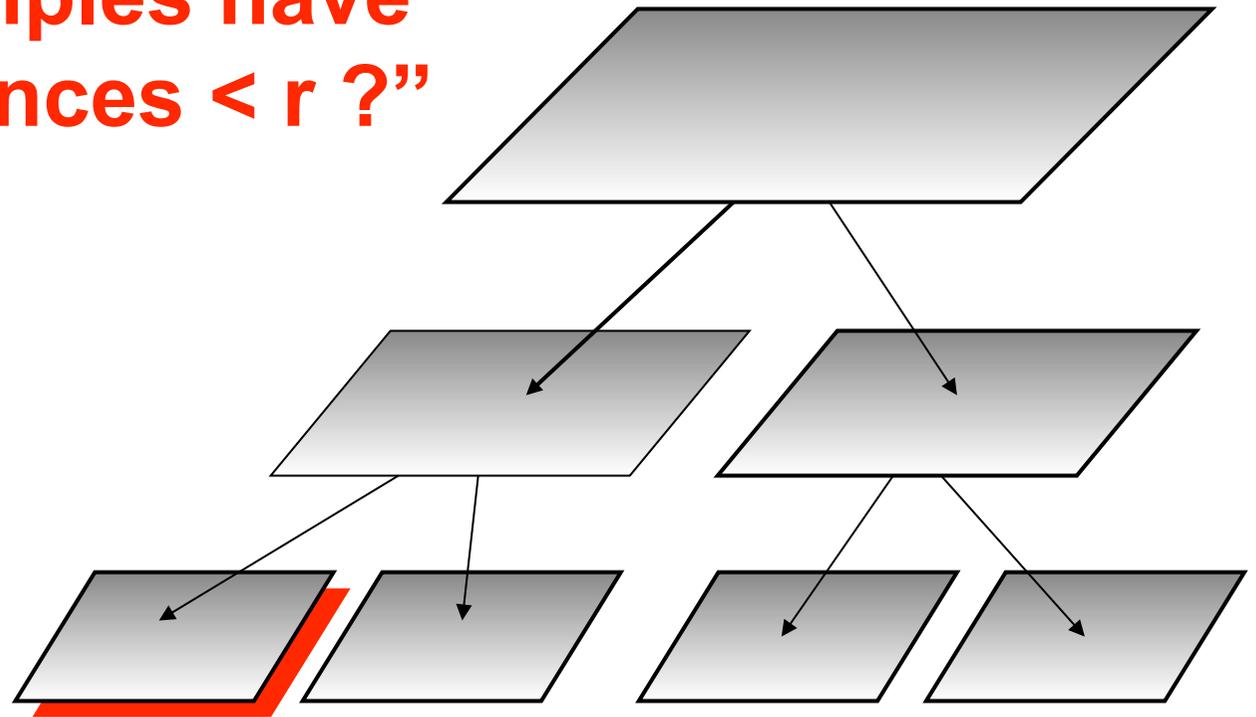
“How many triples have pairwise distances  $< r$  ?”



$$\sum_i^N \sum_{j \neq i}^N \sum_{k \neq j \neq i}^N I(\delta_{ij} < r_1) I(\delta_{jk} < r_2) I(\delta_{ki} < r_3)$$

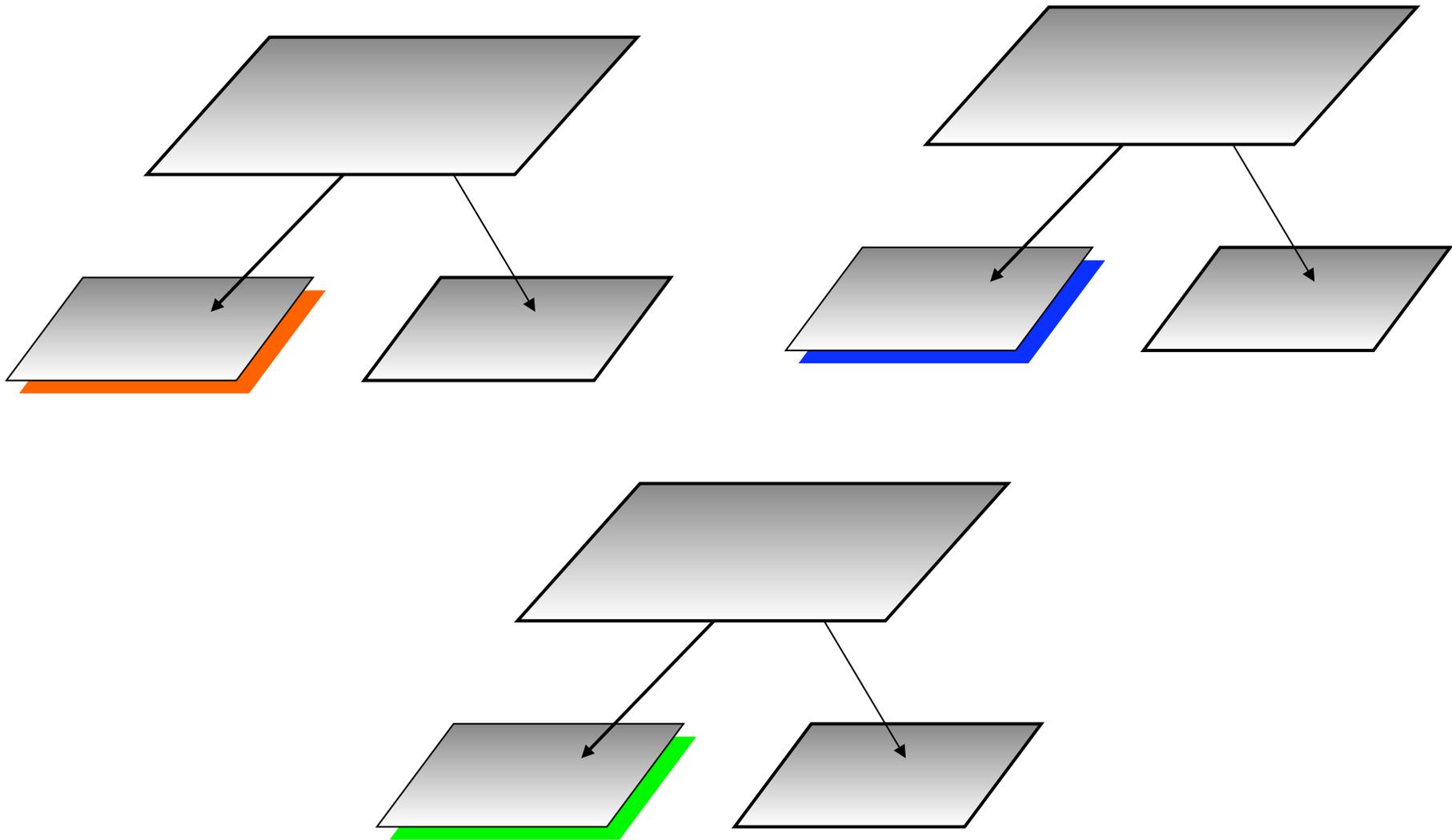
# How can we count $n$ -tuples efficiently?

“How many triples have pairwise distances  $< r$  ?”

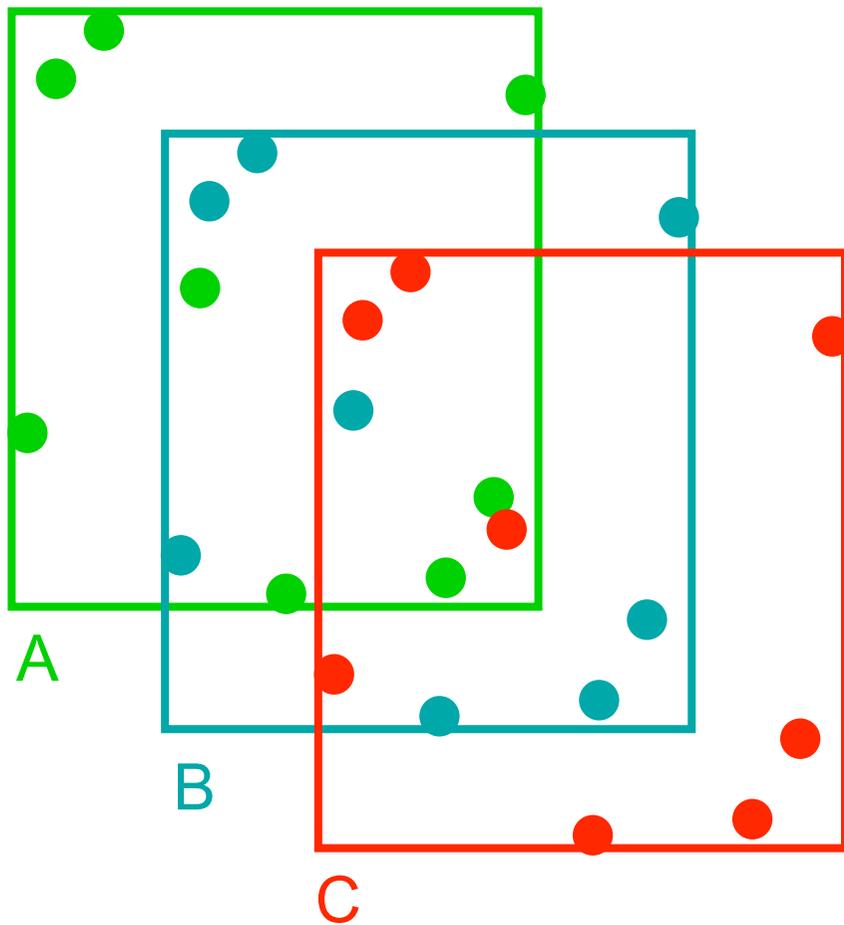


# Use $n$ trees!

[Gray & Moore, NIPS 2000]



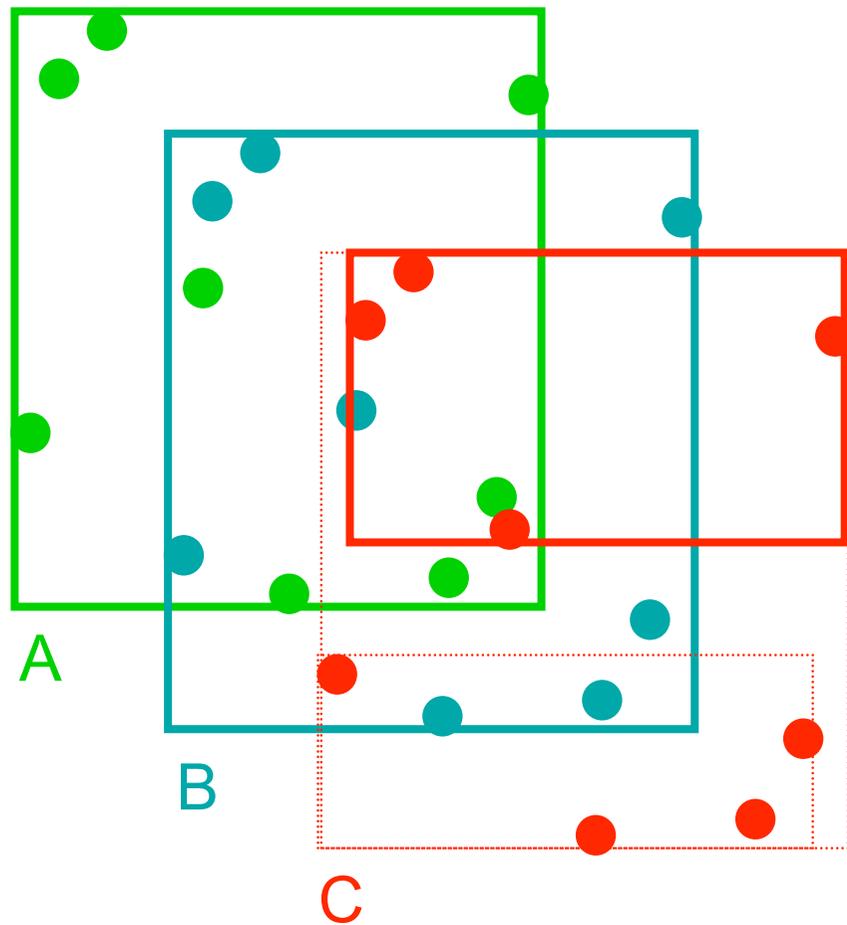
“How many valid triangles  $a$ - $b$ - $c$   
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?



$$\text{count}\{A, B, C\} =$$

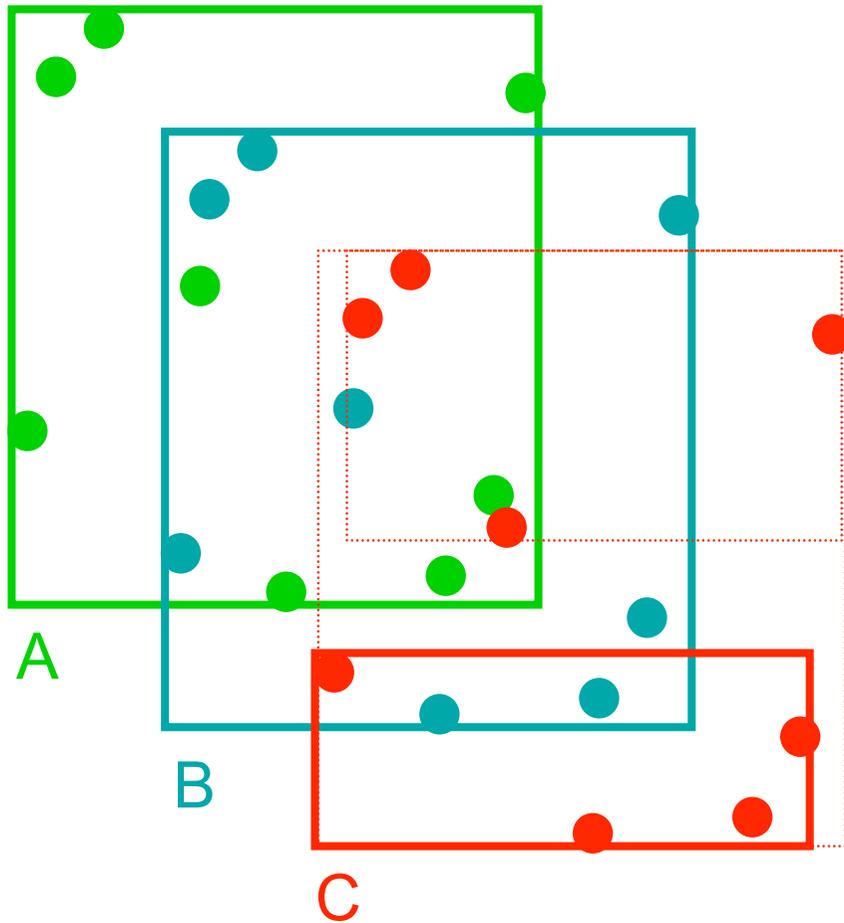
?

“How many valid triangles  $a$ - $b$ - $c$   
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?



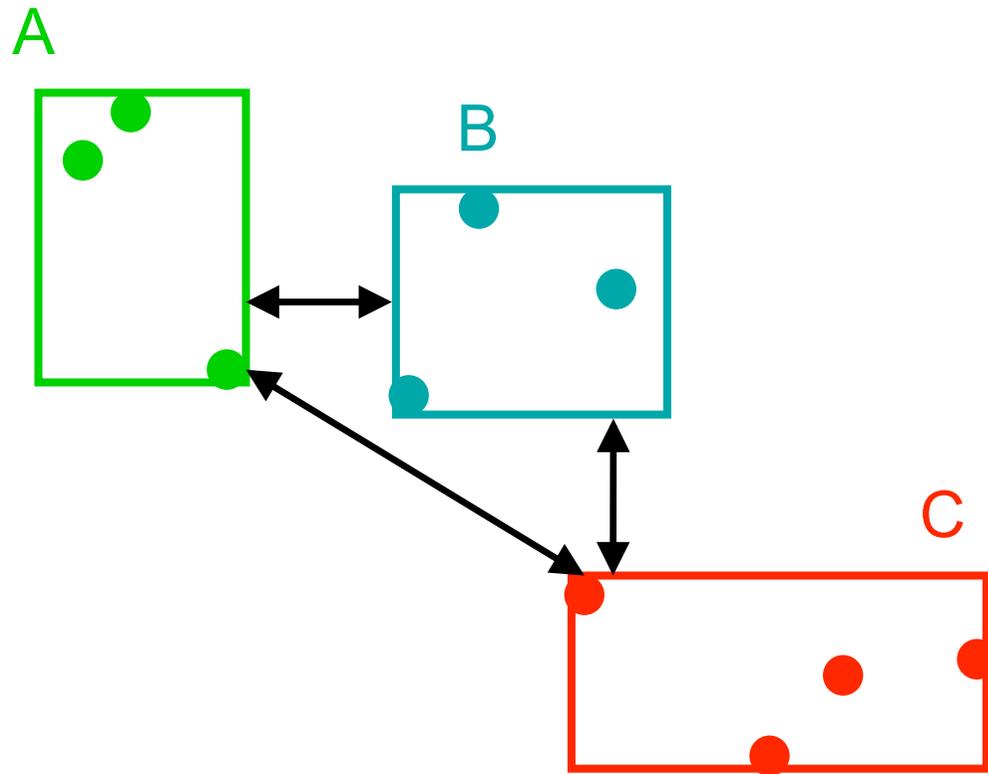
$$\begin{aligned} \text{count}\{A, B, C\} = & \\ \text{count}\{A, B, C.\textit{left}\} & \\ + & \\ \text{count}\{A, B, C.\textit{right}\} & \end{aligned}$$

“How many valid triangles  $a$ - $b$ - $c$   
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?



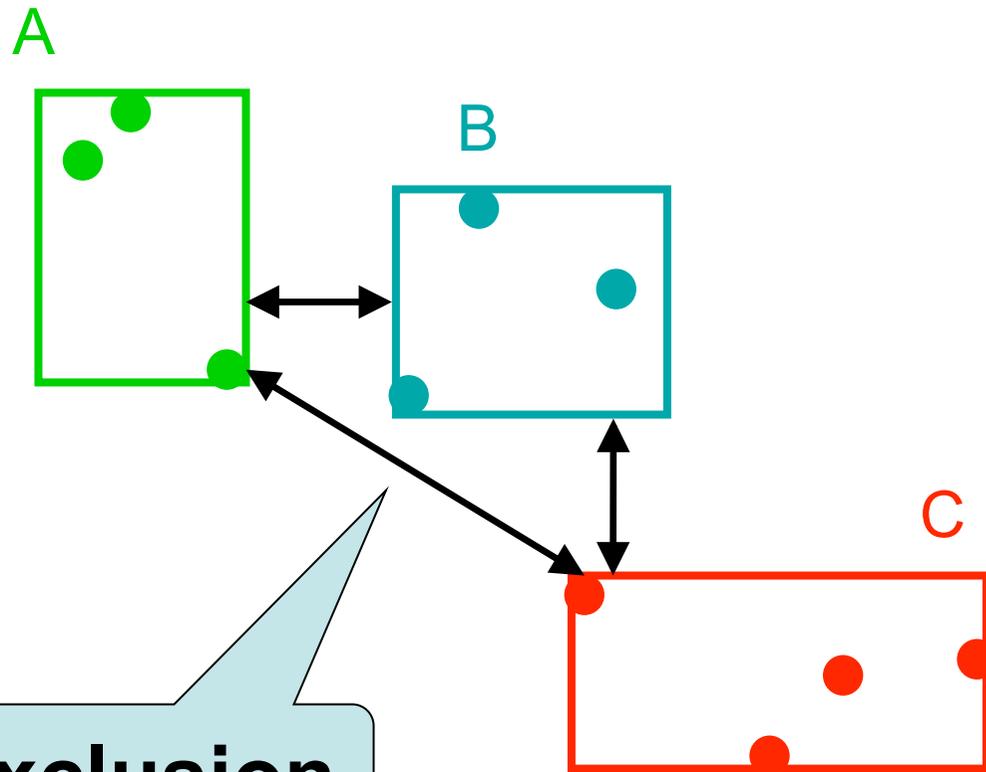
$$\begin{aligned} \text{count}\{A, B, C\} = & \\ \text{count}\{A, B, C.\textit{left}\} & \\ + & \\ \text{count}\{A, B, C.\textit{right}\} & \end{aligned}$$

“How many valid triangles  $a$ - $b$ - $c$   
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?”



count{**A**,**B**,**C**} =  
?

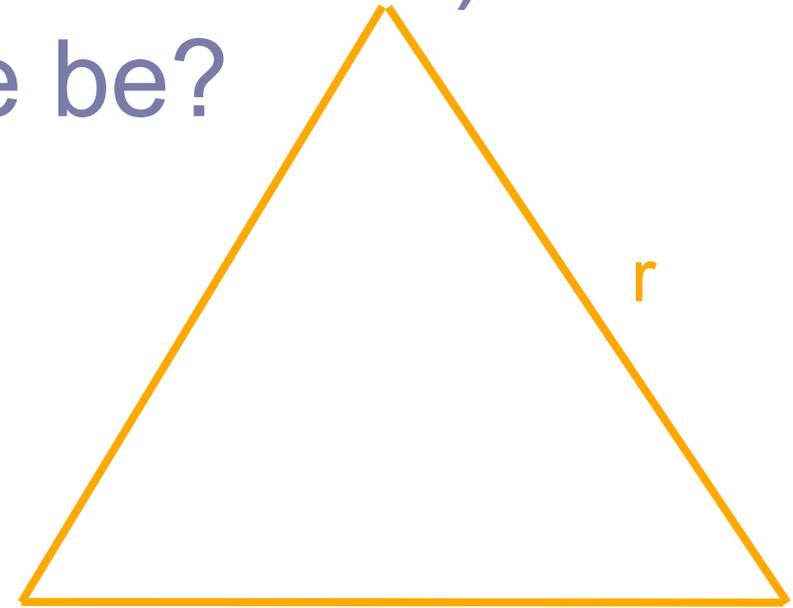
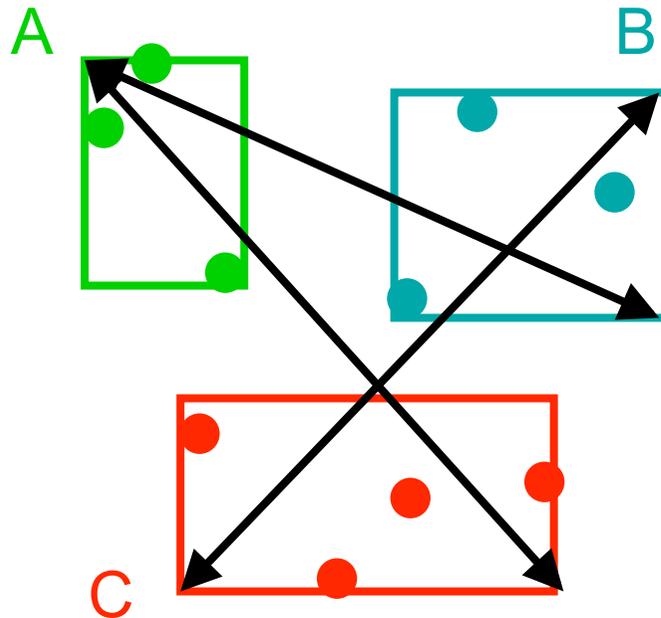
“How many valid triangles a-b-c  
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?



count{A,B,C} =  
**0!**

**Exclusion**

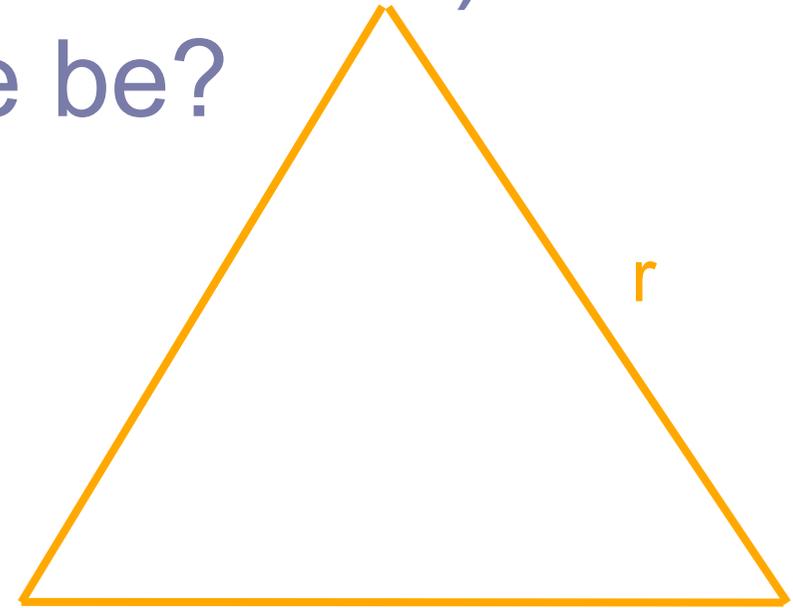
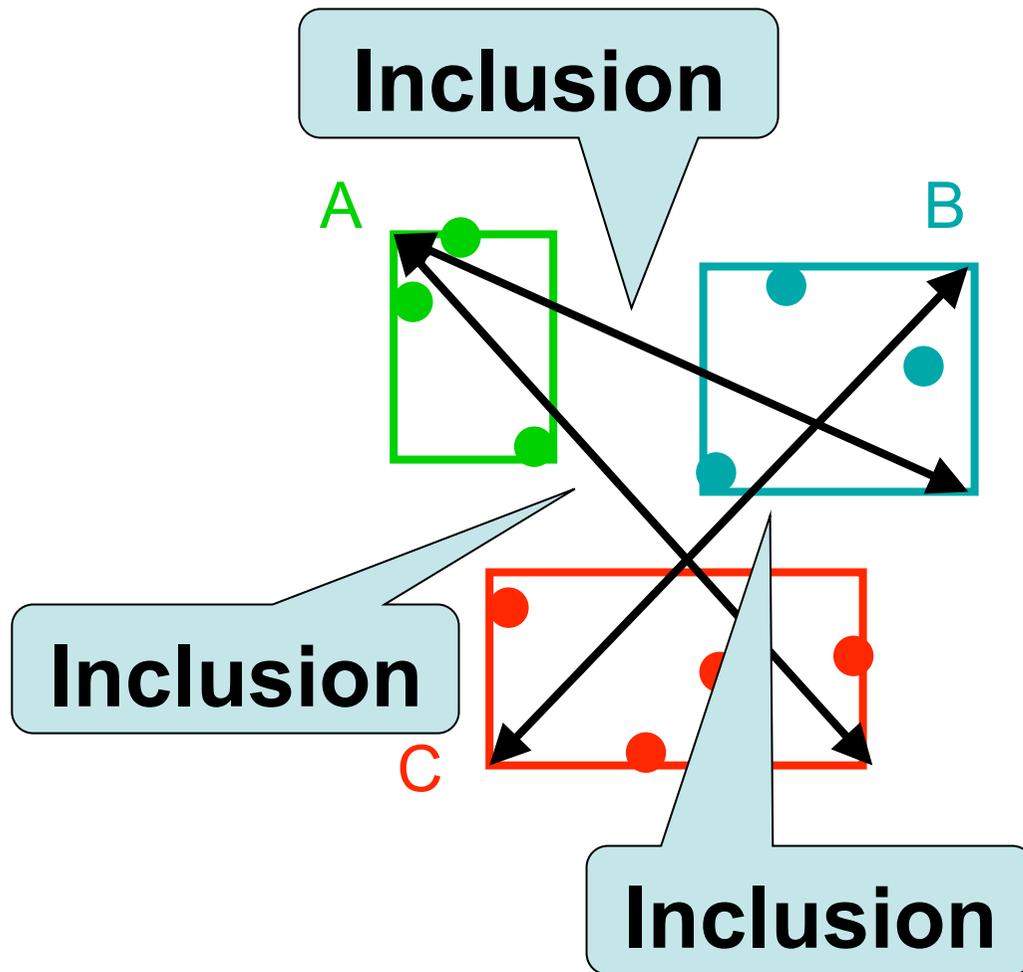
“How many valid triangles a-b-c  
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?”



count{A,B,C} =

?

“How many valid triangles a-b-c  
(where  $a \in A$ ,  $b \in B$ ,  $c \in C$ )  
could there be?”



$$\text{count}\{A, B, C\} =$$

$$|A| \times |B| \times |C|$$

# 3-point runtime

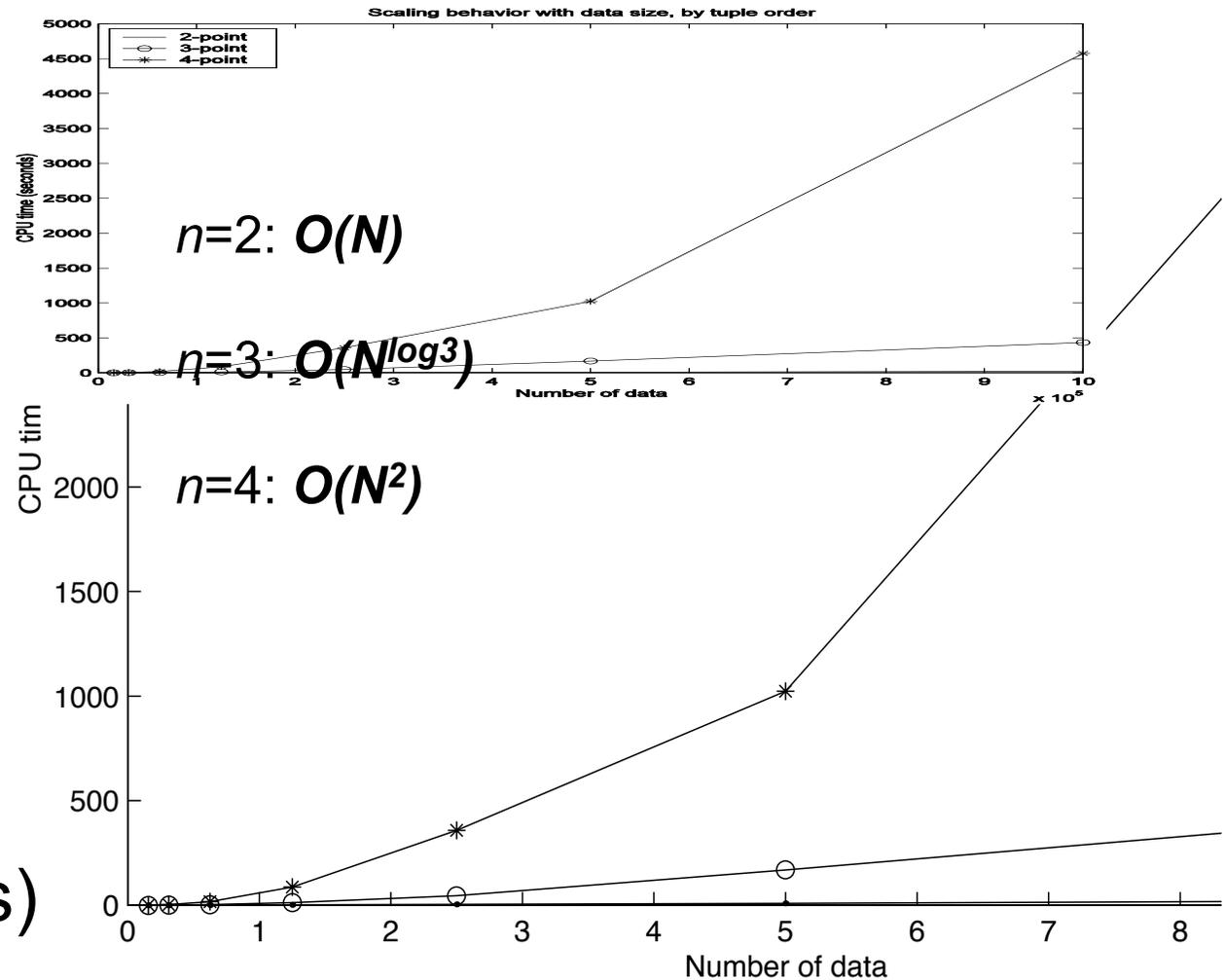
(biggest previous:  
20K)

VIRGO  
simulation data,  
 $N = 75,000,000$

naïve:  $5 \times 10^9$  sec.  
(~150 years)

multi-tree: **55 sec.**

(exact)



# Generalized N-body Problems

- **Interesting approach (for n-point): *n-tree algorithms* [Gray and Moore 2001, NIPS; Moore et al. 2001, Mining the Sky]**
  - First efficient exact algorithm for n-point correlations
- **Interesting approach (for n-point): *Monte Carlo n-tree* [Waters, Riegel and Gray]**
  - Orders of magnitude faster

# Generalized N-body Problems

- **Interesting approach (for EMST): *dual-tree Boruvka algorithm* [March and Gray]**
  - Note this is a cubic problem
- **Interesting approach (N-body decision problems): *dual-tree bounding with hybrid tree expansion* [Liu, Moore, and Gray 2004; Gray and Riegel 2004, CompStat; Riegel and Gray 2007, SDM]**
  - An exact classification algorithm

# Generalized N-body Problems

- **Interesting approach (Gaussian kernel):**  
*dual-tree with multipole/Hermite expansions* [Lee, Gray and Moore 2005, NIPS; Lee and Gray 2006, UAI]
  - Ultra-accurate fast kernel summations
- **Interesting approach (arbitrary kernel):**  
*automatic derivation of hierarchical series expansions* [Lee and Gray]
  - For large class of kernel functions

# Generalized N-body Problems

- **Interesting approach (summative forms): *multi-scale Monte Carlo* [Holmes, Gray, Isbell 2006 NIPS; Holmes, Gray, Isbell 2007, UAI]**
  - Very fast bandwidth learning
- **Interesting approach (summative forms): *Monte Carlo multipole methods* [Lee and Gray 2008, NIPS]**
  - Uses SVD tree

# Generalized N-body Problems

- **Interesting approach (for multi-body potentials in physics): *higher-order multipole methods* [Lee, Waters, Ozakin, and Gray, et al.]**
  - First fast algorithm for higher-order potentials
- **Interesting approach (for quantum-level simulation): *4-body treatment of Hartree-Fock* [March and Gray, et al.]**

# Graphical model inference

- **How it appears:** hidden Markov models, bipartite matching, Gaussian and discrete graphical models
- **Common methods:** belief propagation, expectation propagation
- **Mathematical challenges:** large cliques, upper and lower bounds, graphs with loops, parallel
- **Mathematical topics:** variational methods, statistical physics, turbo codes

# Graphical model inference

- **Interesting method (for discrete models):**  
*Survey propagation [Mezard et al 2002]*
  - *Good results for combinatorial optimization*
  - *Based on statistical physics ideas*
- **Interesting method (for discrete models):**  
*Expectation propagation [Minka 2001]*
  - *Variational method based on moment-matching idea*
- **Interesting method (for Gaussian models):**  $L_p$   
*structure search, solve linear system for inference [Tran, Lee, Holmes, and Gray]*

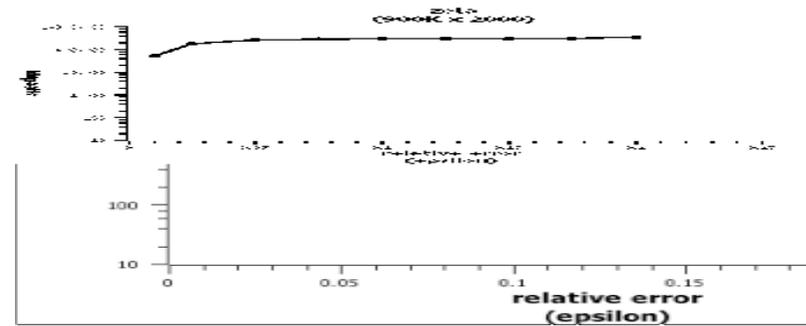
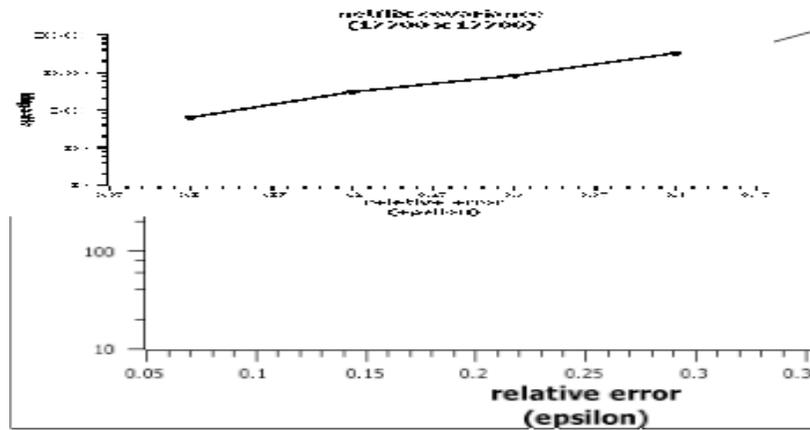
# Linear algebra

- **How it appears:** linear regression, Gaussian process regression, PCA, kernel PCA, Kalman filter
- **Common methods:** QR, Krylov, ...
- **Mathematical challenges:** numerical stability, sparsity preservation, ...
- **Mathematical topics:** linear algebra, randomized algorithms, Monte Carlo

# Linear algebra

- **Interesting method (for probably-approximate k-rank SVD): Monte Carlo k-rank SVD [Frieze, Drineas, et al. 1998-2008]**
  - *Sample either columns or rows, from squared length distribution*
  - *For rank-k matrix approx; must know k*
- **Interesting method (for probably-approximate full SVD): QUIC-SVD [Holmes, Gray, Isbell 2008, NIPS]; QUIK-SVD [Holmes and Gray]**
  - *Sample using cosine trees and stratification*
  - *Builds tree as needed*
  - *Full SVD: automatically sets rank based on desired error*

# QUIC-SVD speedup



38 days  $\rightarrow$  1.4 hrs, 10% rel. error

40 days  $\rightarrow$  2 min, 10% rel. error

# Optimization

- **How it appears:** support vector machine, maximum variance unfolding, robust  $L_2$  estimation
- **Common methods:** interior point, Newton's method
- **Mathematical challenges:** ML-specific objective functions, large number of variables / constraints, global optimization, parallel
- **Mathematical topics:** optimization theory, linear algebra, convex analysis

# Optimization

- **Interesting method:** *Sequential minimization optimization (SMO) [Platt 1999]*
  - Much more efficient than interior-point, for SVM QPs
- **Interesting method:** *Stochastic quasi-Newton [Schraudolf 2007]*
  - Does not require scan of entire data
- **Interesting method:** *Sub-gradient methods [Vishwanathan and Smola 2006]*
  - Handles kinks in regularized risk functionals
- **Interesting method:** *Approximate inverse preconditioning using QUIC-SVD for energy minimization and interior-point [March, Vasiloglou, Holmes, Gray]*
  - Could potentially treat a large number of optimization problems

# Now fast!

very fast as fast as possible (conjecture)

- **Querying:** nearest-neighbor, sph range-search, ortho range-search, all-nn
- **Density estimation:** kernel density estimation, mixture of Gaussians
- **Regression:** linear regression, kernel regression, Gaussian process regression
- **Classification:** nearest-neighbor classifier, nonparametric Bayes classifier, support vector machine
- **Dimension reduction:** principal component analysis, non-negative matrix factorization, kernel PCA, maximum variance unfolding
- **Outlier detection:** by robust  $L_2$  estimation
- **Clustering:** k-means, mean-shift, hierarchical clustering (“friends-of-friends”)
- **Time series analysis:** Kalman filter, hidden Markov model, trajectory tracking
- **Feature selection and causality:** LASSO regression,  $L_1$  support vector machine, Gaussian graphical models, discrete graphical models
- **2-sample testing and matching:** n-point correlation, bipartite matching

# Astronomical applications

- **All-k-nearest-neighbors:  $O(N^2) \rightarrow O(N)$ , exact.** Used in *[Budavari et al., in prep]*
- **Kernel density estimation:  $O(N^2) \rightarrow O(N)$ , rel err.** Used in *[Balogh et al. 2004]*
- **Nonparametric Bayes classifier (KDA):  $O(N^2) \rightarrow O(N)$ , exact.** Used in *[Richards et al. 2004,2009], [Scranton et al. 2005]*
- **n-point correlations:  $O(N^n) \rightarrow O(N^{\log n})$ , exact.** Used in *[Wake et al. 2004], [Giannantonio et al 2006],[Kulkarni et al 2007]*

# Astronomical highlights

- **Dark energy** evidence, *Science* 2003, Top Scientific Breakthrough of the year (n-point)
  - 2007 biggest 3-point calculation ever
- **Cosmic magnification** verification *Nature* 2005 (nonparam. Bayes clsf)
  - 2008 largest quasar catalog ever

# A few others to note...

very fast as fast as possible (conjecture)

- **Querying:** nearest-neighbor, sph range-search, ortho range-search, all-nn
- **Density estimation:** kernel density estimation, mixture of Gaussians
- **Regression:** linear regression, **kernel regression**, **Gaussian process regression**
- **Classification:** nearest-neighbor classifier, nonparametric Bayes classifier, support vector machine
- **Dimension reduction:** **principal component analysis**, non-negative matrix factorization, kernel PCA, maximum variance unfolding
- **Outlier detection:** by robust  $L_2$  estimation
- **Clustering:** k-means, mean-shift, **hierarchical clustering** (“friends-of-friends”)
- **Time series analysis:** Kalman filter, **hidden Markov model**, trajectory tracking
- **Feature selection and causality:** LASSO regression,  $L_1$  support vector machine, Gaussian graphical models, discrete graphical models
- **2-sample testing and matching:** n-point correlation, bipartite matching

# How to do Machine Learning on Massive Astronomical Datasets?

1. Choose the appropriate **statistical task and method** for the scientific question
2. Use the fastest **algorithm and data structure** for the statistical method
3. Put it in **software**

# Keep in mind the machine

- **Memory hierarchy:** cache, RAM, out-of-core
- Dataset bigger than one machine: **parallel/distributed**
- Everything is becoming **multicore**
- **Cloud computing:** software as a service

# Keep in mind the overall system

- **Databases** can be more useful than ASCII files (e.g. CAS)
- **Workflows** can be more useful than brittle perl scripts
- **Visual analytics** connects visualization/HCI with data analysis (e.g. In-SPIRE)

# Our upcoming products

- **MLPACK**: “the LAPACK of machine learning” – Dec. 2008 [**FASTlab**]
- **THOR**: “the MapReduce of Generalized N-body Problems” – Apr. 2009 [**Boyer, Riegel, Gray**]
- **CAS Analytics**: fast data analysis in CAS (SQL Server) – Apr. 2009 [**Riegel, Aditya, Krishnaiah, Jakka, Karnik, Gray**]
- **LogicBlox**: all-in-one business intelligence [**Kanetkar, Riegel, Gray**]

# Keep in mind the software complexity

- Automatic **code generation** (e.g. MapReduce)
- Automatic **tuning** (e.g. OSKI)
- Automatic **algorithm derivation** (e.g. SPIRAL, AutoBayes) [**Gray et al. 2004; Bhat, Riegel, Gray, Agarwal**]

# The end

- We have/will have fast algorithms for most data analysis methods in MLPACK
- Many opportunities for applied math and computer science in large-scale data analysis
- Caveat: Must treat the right problem
- Computational astronomy workshop and large-scale data analysis workshop coming soon

Alexander Gray [agray@cc.gatech.edu](mailto:agray@cc.gatech.edu)

(email is best; webpage sorely out of date)