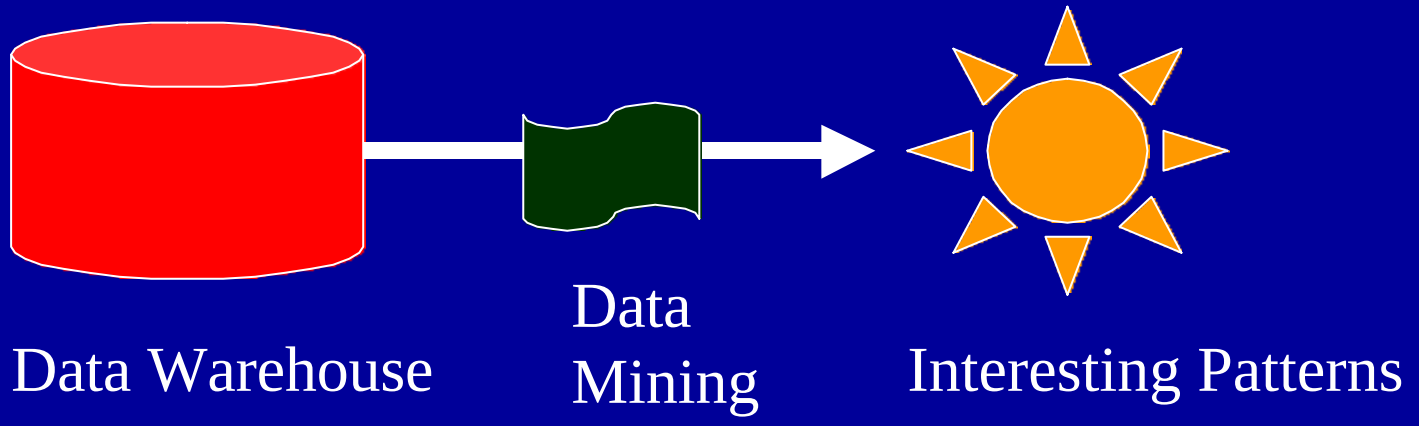
Incompleteness in Data Mining

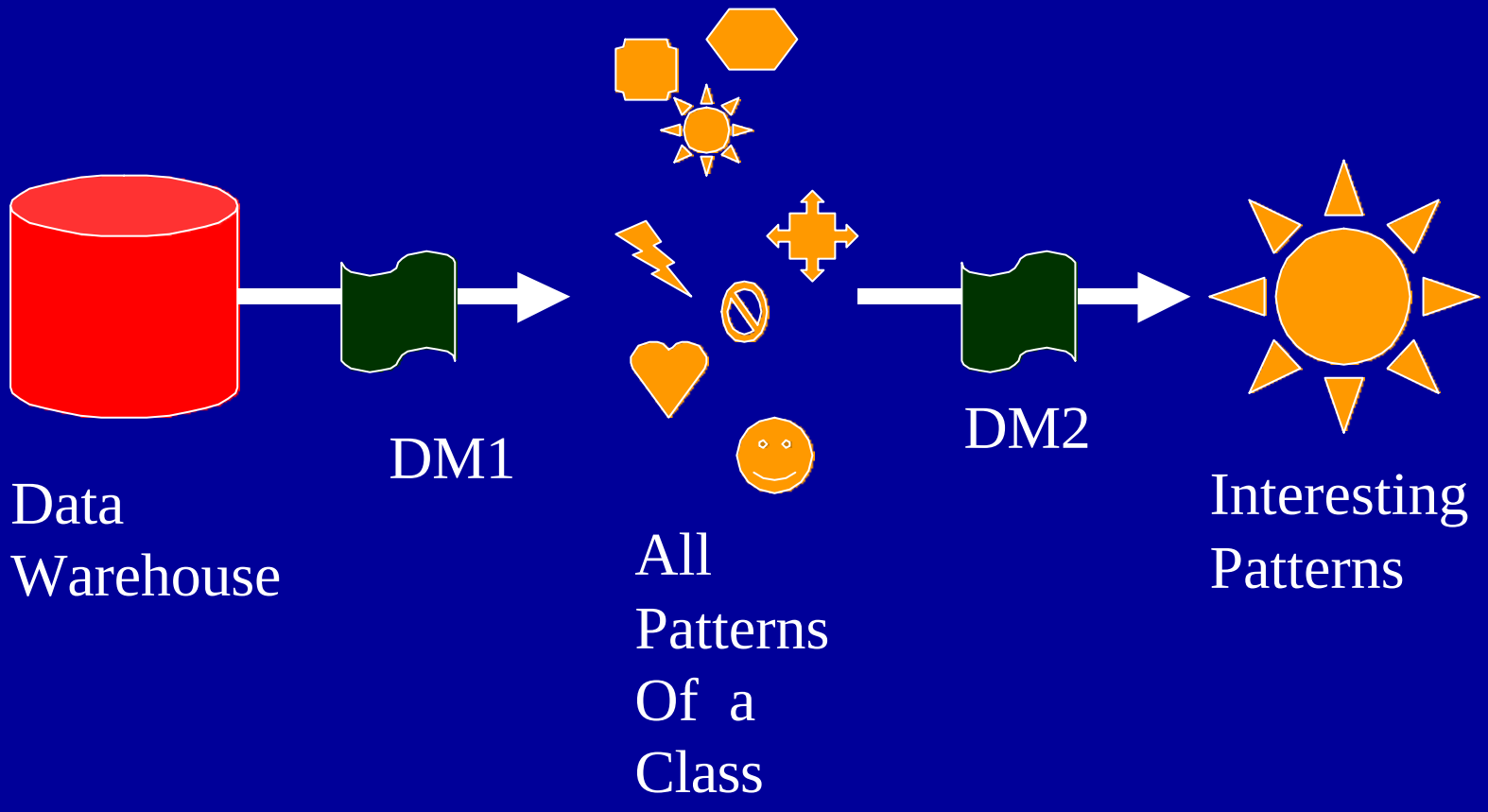
H. V. Jagadish
University of Michigan

(with generous help from Raymond Ng)

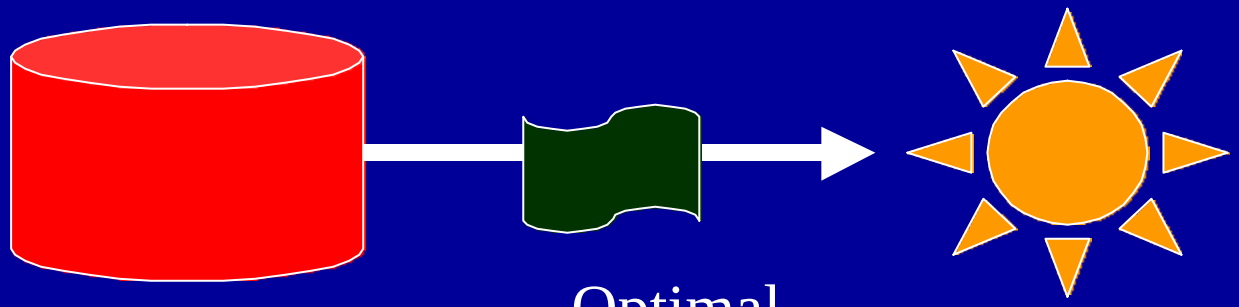
The Data Mining Ideal



Data Mining in Practice



Data Mining As Optimization



What is Interesting?

- = Best ?
  - Quantitative metric of optimality as surrogate for the non-measurable
- Leads to an exhaustive search for ALL possibly interesting patterns.

Perfection Has a Price

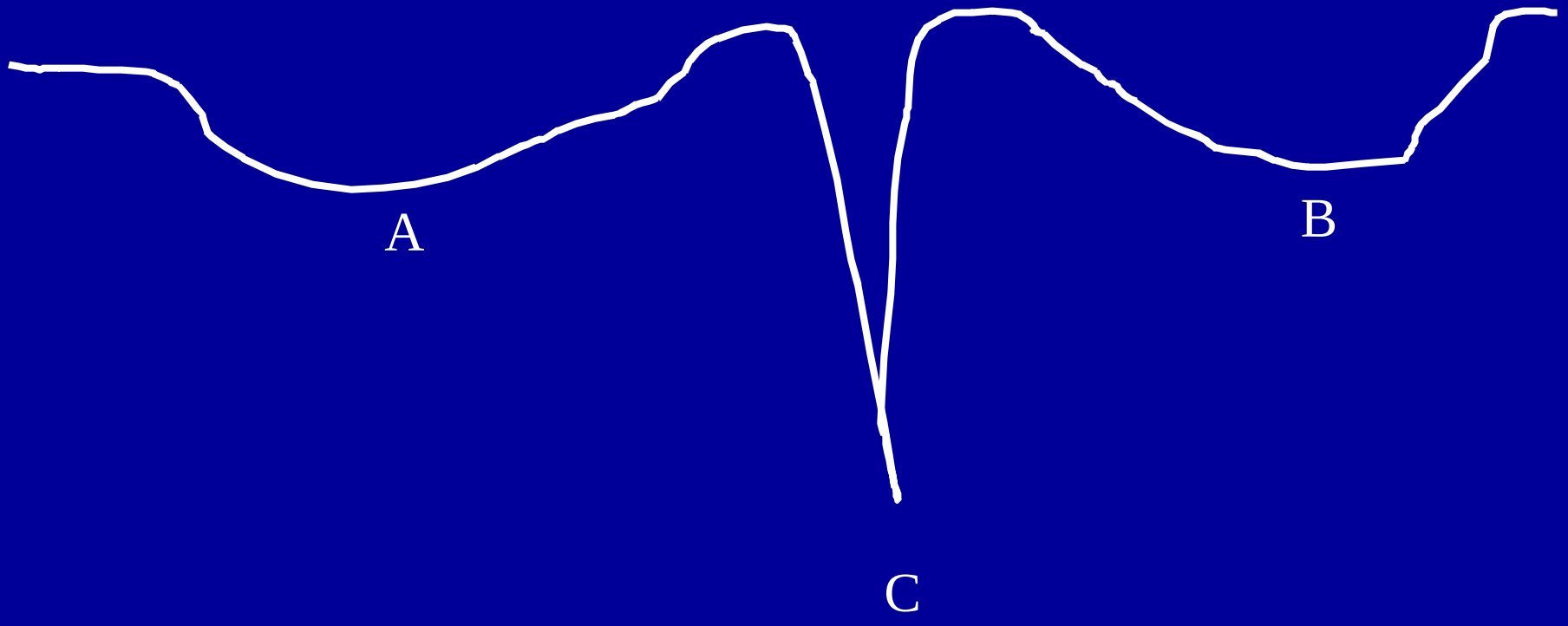
- Often require combinatorial optimization, with exhaustive consideration of an exponential number of options.
- To render algorithms tractable, the optimization is often defined over problems that are too narrowly constrained.

Our Thesis

- One should use cheap approximate techniques to do almost the best, and to cover almost all the possibilities.

- Approximation heuristics are not very good for many difficult optimization problems, BUT, by their nature, data mining problems are “special”

Intuition



The Human in the Loop

- For OLAP, we now have a well-accepted need for data reduction, and for approximate query answering.
- If data mining is to be an interactive process, we need tools for quick and dirty data mining.

An Example

- Finding **fascicles.**
  - Interesting patterns in subsets of a relation.
  - Introduced in JMN_VLDB99.
- Pattern is a set of attributes on which the tuples (almost) match.

Employee Table

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

Some Fascicles

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

More Fascicles

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

Even More Fascicles

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

Some Fascicles with k=1

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

More Fascicles with k=1

EMPLOYEE	AGE	SALARY	HOURS
Alice	24	55	40
Bob	25	52	40
Carlos	31	41	40
Dmitri	30	66	20
Elena	30	42	20
Faroukh	26	40	40
Girija	52	43	40
Hillary	29	64	40

Association Rule Mining

- Much studied data mining problem
- Given a number of transactions, in each of which several items were purchased ------- find sets of items that were frequently purchased together.

E.g. AC is frequent (>=5) in:

ABC, ACD, BCE, ACF, BCG, ABCD, ACDE

A Priori Algorithm

- Exponential number of item combinations – cannot count them all.
- Count candidate combinations of size k in the k^{th} pass.
- Prune candidates using anti-monotonicity: An item set can be frequent only if each of its subsets is frequent.

Like Association Rules?

- Bin attribute values into discrete buckets.
- Call each distinct attribute value occurrence an “item”.
- Each tuple in the relation is an item-set.
- A fascicle with k compact attributes is a frequent itemset of size k.
- Compute using A Priori, or variants.

Randomized Algorithm

- Pick a tuple at random.
- Pick a second tuple at random, and see if at least k attributes match.
- If so, pick a third tuple at random.
- Keep going until adding one more tuple leaves less than k compact attributes.
- Select all tuples in the relation with these k attribute values in the compact range.

Performance -- Runtime

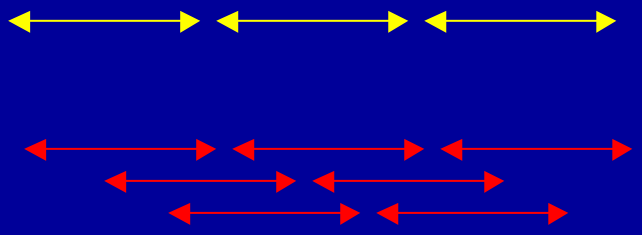
k	Randomized	A Priori
2	3.5s	13.5s
3	3.5s	299.8s
4	3.4s	1591.0s
5	3.4s	> 2500s

Performance -- Coverage

k	Randomized	A Priori
2	0.999	0.992
3	0.988	0.967
4	0.974	0.846
5	0.941	N/A

Poor Coverage in A Priori

- Due to “pre-binning”.
- Coverage can be improved, at much greater computational cost by creating overlapping bins.



Lesson Learned

- Approximate algorithms can not only solve problems (at least for the one example problem considered) a lot faster than optimal algorithms, but also:
- They are amenable to a more general problem statement and hence can produce better overall results than a narrowly constructed optimal algorithm.

Desiderata

Approximate/Interactive Algorithms must:

- Be quantifiable, w.r.t. error guarantees;
- Be tunable, w.r.t. degree of error;
- Be incremental, w.r.t. finding solutions to related problems;

Lots of interesting research questions …

In Conclusion

- Much good work in approximate query answering for decision support.
- Some excellent work on sample-based clustering, sampling for association rule mining.
- But much yet to be done before we can reach the holy grail of interactive data mining.