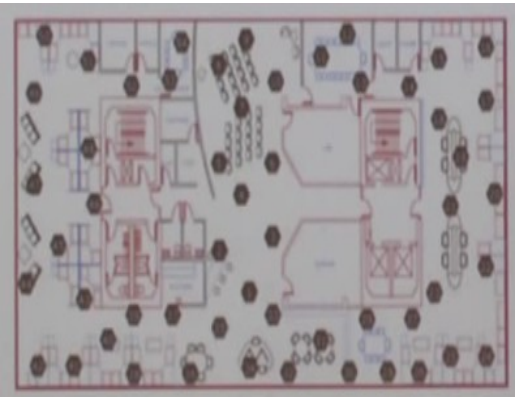
Scorpion, a detective that is good at explaining the anomalies

Scorpion,
explaining **outliers in aggregate queries**

- Aggregation function: SUM, AVERAGE, STD, MIN,MAX

outliers in aggregate queries



Time	SensorID	Voltage	Humidity	Temp.
11AM	1	2.64	0.4	34
11AM	2	2.65	0.5	35
11AM	3	2.63	0.4	35
12PM	1	2.7	0.3	35
12PM	2	2.7	0.5	35
12PM	3	2.3	0.4	100
1PM	1	2.7	0.3	35
1PM	2	2.7	0.5	35
1PM	3	2.3	0.5	80

A table that records the attributes of sensors

SELECT **avg**(temp),time
FROM sensors GROUP BY time

Time	AVG(temp)	Label
11AM	34.6	Hold-out
12PM	56.6	Outlier
1PM	50	Outlier

WHY?

outliers in aggregate queries

Dataset from A hospital:
A table with one row per patient visit, and 45 columns that describe patient demographics, diagnoses, and other attributes describing the visit

```
SELECT SUM(expense), disease
FROM hospital GROUP BY disease
```

lung cancer cases disproportionately account for millions of dollars

WHY?

Scorpion :

an interactive system answer “why” questions in the context of SQL aggregation queries.

Time	SensorID	Voltage	Humidity	Temp.
11AM	1	2.64	0.4	34
11AM	2	2.65	0.5	35
11AM	3	2.63	0.4	35
12PM	1	2.7	0.3	35
12PM	2	2.7	0.5	35
12PM	3	2.3	0.4	100
1PM	1	2.7	0.3	35
1PM	2	2.7	0.5	35
1PM	3	2.3	0.5	80

Time	AVG(temp)	Label
11AM	34.6	Hold-out
12PM	56.6	Outlier
1PM	50	Outlier

individual tuples are **less informative**, not enough for understanding why

Scorpion tells why:

Volt < 2.5 & Sensor = 3

predicate

- Predicates give broader, coarse-grained explanations
- Predicates: Conjunction of discrete subsets and continuous range

Volt < 2.5 & Sensor = 3

Conjunction

Sensor
Sensor &Voltage
Sensor&Voltage&Light
exponential in
# of attributes

Discrete

Sensor=1
Sensor= 1 or 2
Sensor=2 or 3
exponential in
# of unique values

Continuous

Sensor=1
Sensor= 1 or 2
Sensor=2 or 3
Quadratic in # of
unique values

↓

Predicate space is very large

scoring function

We need a scoring function for ranking and returning to p explanations in the exponential space

Scoring Function: Influence(P)

- Quantify how much predicate P influenced the result of aggregation function (sum,average)

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\#\text{ of records to make the change})}$$

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\#\text{ of records to make the change})}$$

12PM	1	2.7	0.3	35
12PM	2	2.7	0.5	35
12PM	3	2.3	0.4	100

Predicate1: sensor=3
One tuple causes the change

infl(P1)=(56.5-35)/1=21.6

Predicate2: sensor=3 or 2
Two tuples cause the change
infl(P2)=21.6/2=10.8

Predicate3: sensor=3 or 2 or 1
Three tuples cause the change
infl(P3)=51.6/3=17.2

$$\text{infl}_{agg}(p) = \frac{\text{Change in output}}{(\#\text{ of records to make the change})^{\lambda}}$$

Lamda is a hyperparameter : Leave the choice to the user

Lamda=1

Predicate1: sensor=3
One tuple causes the change

Predicate2: sensor=3 or 2
Two tuples cause the change

lamda=0

Predicate3: sensor=3 or 2 or 1
Three tuples cause the change

The higher the lamda, the more selective predicate it produces

$$infl_{agg}(p) = \frac{\text{Change in output} \times V}{(\#\text{ of records to make the change})^{\lambda}}$$

V: users indicate whether the outliers are too high or too low

Time	AVG(temp)	Label	v
11AM	34.6	Hold-out	-
12PM	56.6	Outlier	< +1 >
1PM	50	Outlier	< +1 >

12PM	1	2.7	0.3	35
12PM	2	2.7	0.5	35
12PM	3	2.3	0.4	100

Predicate1: sensor=3
Infl(P1)=26.1*1=26.1
Predicate2:sensor=1
Infl(P2)=-10.9*1=-10.9

Predicate1: sensor=3
Infl(P1)=26.1*(-1)=-26.1
Predicate2:sensor=1
Infl(P2)=-10.9*(-1)=10.9

$$inf_{agg}(o, h, p, v_o) = C\, inf_{agg}(o, p, v_o) - (1 - C)|inf_{agg}(h, p)|$$

C: users indicate how much the predicate should take the hold- out results into consideration

Time	AVG(temp)	Label	v
11AM	34.6	Hold-out	-
12PM	56.6	Outlier	< +1 >
1PM	50	Outlier	< +1 >

$$inf_{agg}(O, H, p, V) = \lambda \frac{1}{|O|} \sum_{o \in O} inf_{agg}(o, p, v_o) - (1 - \lambda) \max_{h \in H} |inf_{agg}(h, p)|$$

The user often select multiple outliers results and hold-out results, we extend the notion by averaging the influence over the outlier results

Time	AVG(temp)	Label	v
11AM	34.6	Hold-out	-
12PM	56.6	Outlier	< +1 >
1PM	50	Outlier	< +1 >

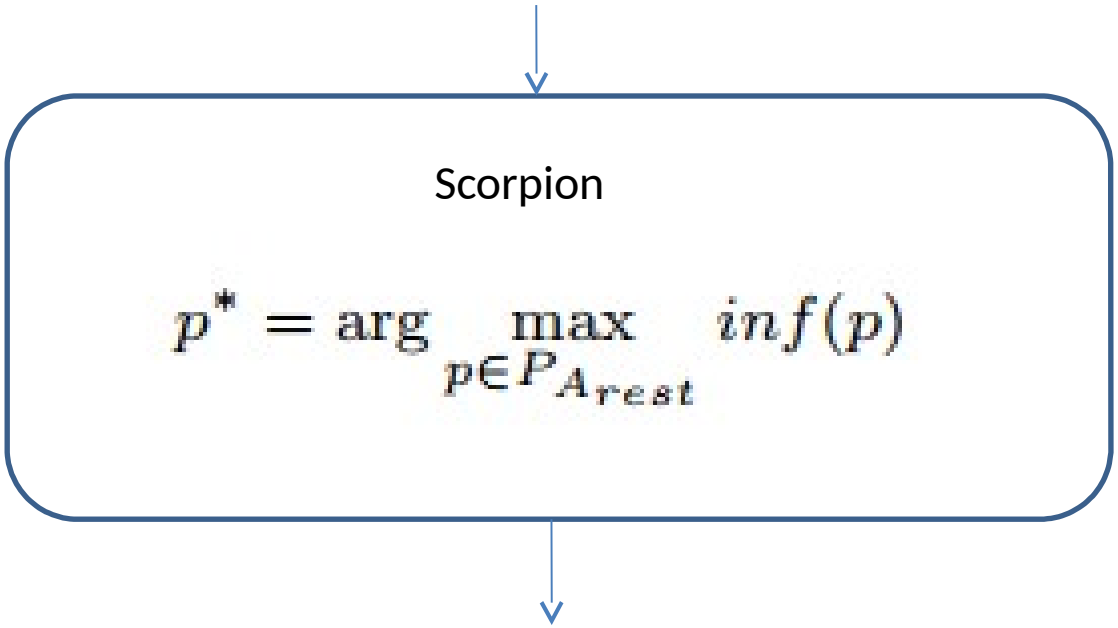
Influential Predicates Problem

$$p^* = \arg \max_{p \in P_{A_{rest}}} inf(p)$$

Scorpion Architecture

Input:

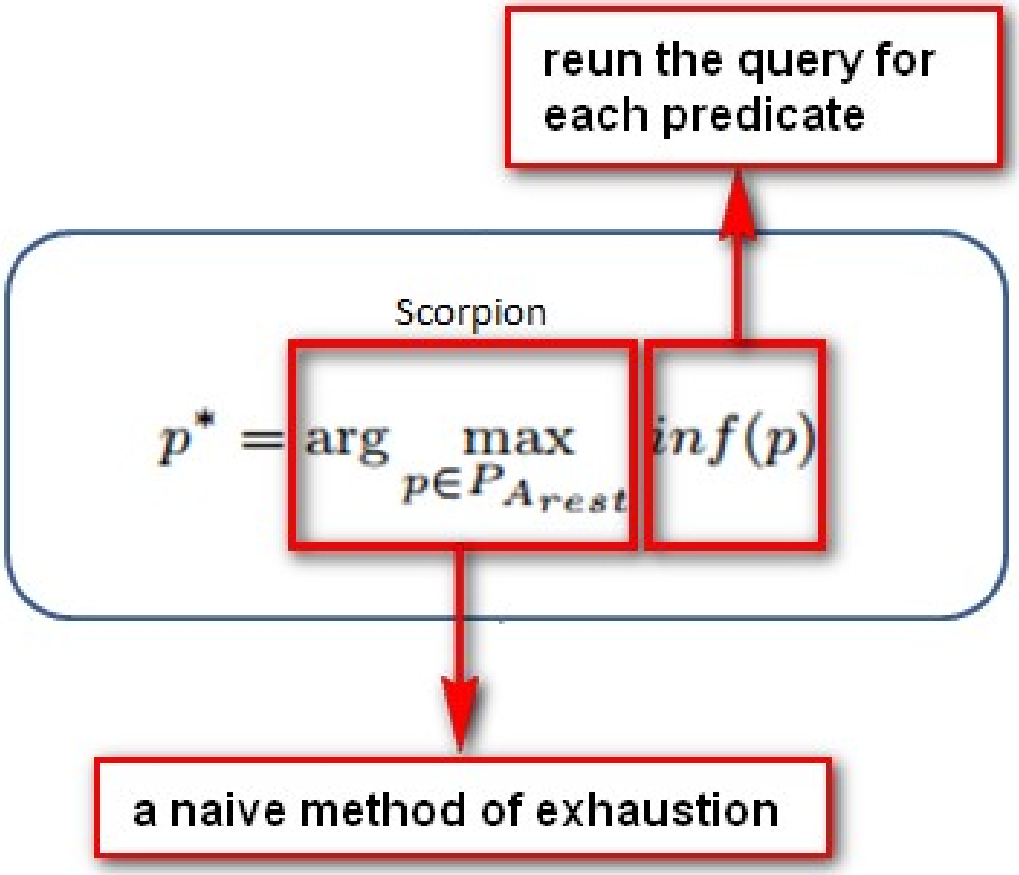
SQL query, outliers, hold-out results, λ, v, C



Output :

predicate p having highest influence

A naïve implementation



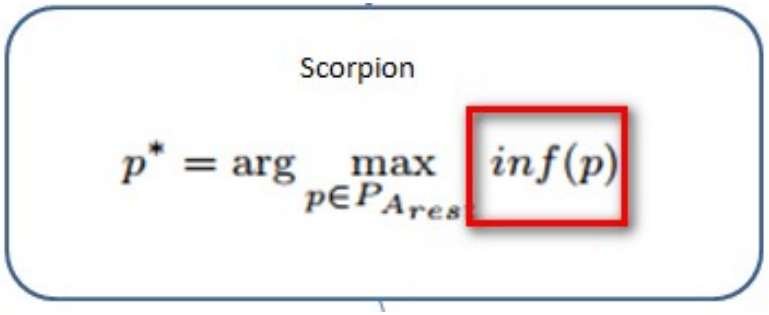
Scorpion

$$p^* = \arg \max_{p \in P_{A_{rest}}} inf(p)$$

Explore some properties of aggregation function → Enable more efficient implementations

some properties

- **Incrementally removable**



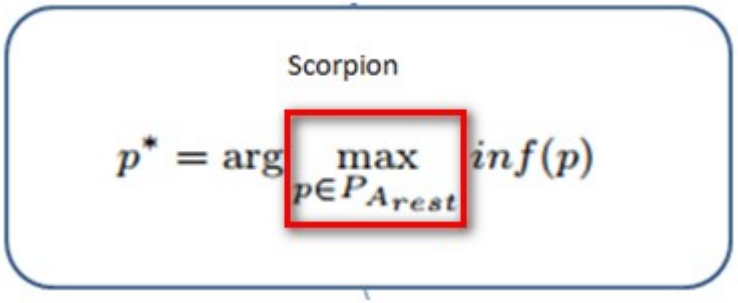
Influ(P)=15-SUM({1,2,3})

Influ(P)=15-(15-SUM({4,5}))=SUM({4,5})

SUM, AVG,STD, COUNT, ~~MAX,MIN~~

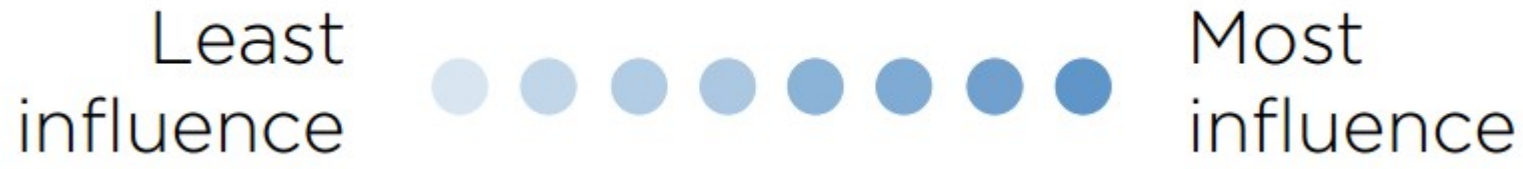
some properties

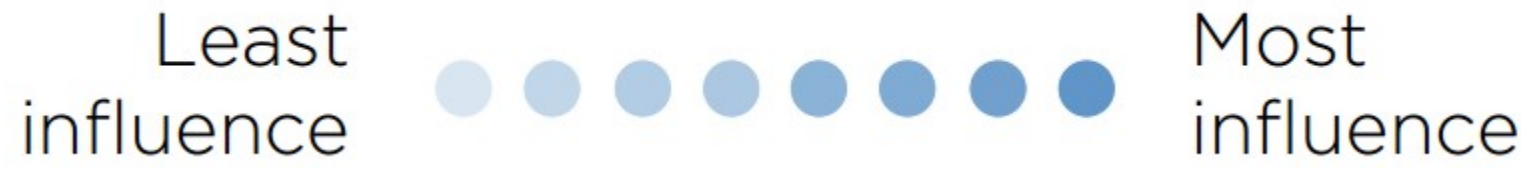
- Incrementally removable
- **Independent**



1.the influence of a set of tuples strictly depends on the influen ces of the indi vidual tuples

2. adding a tuple t more influential than t $*$ $\in$ T with minimum influence ca n't decrease the set's influence

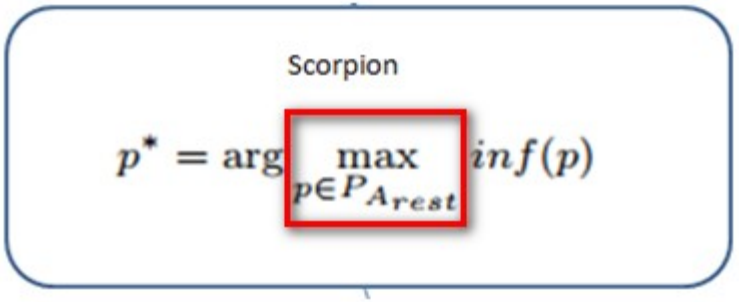




TOP-DOWN decision tree-based algorithm that recursively partitions the predicates and merges similar(adjacent) predicates

some properties

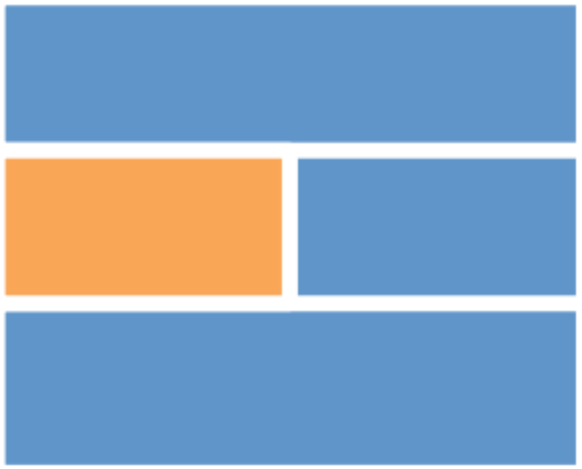
- Incrementally removable
- Independent
- Anti-monotonic



Non-influential tuple sets can not contain influential sub-tuple sets: prune the search space!

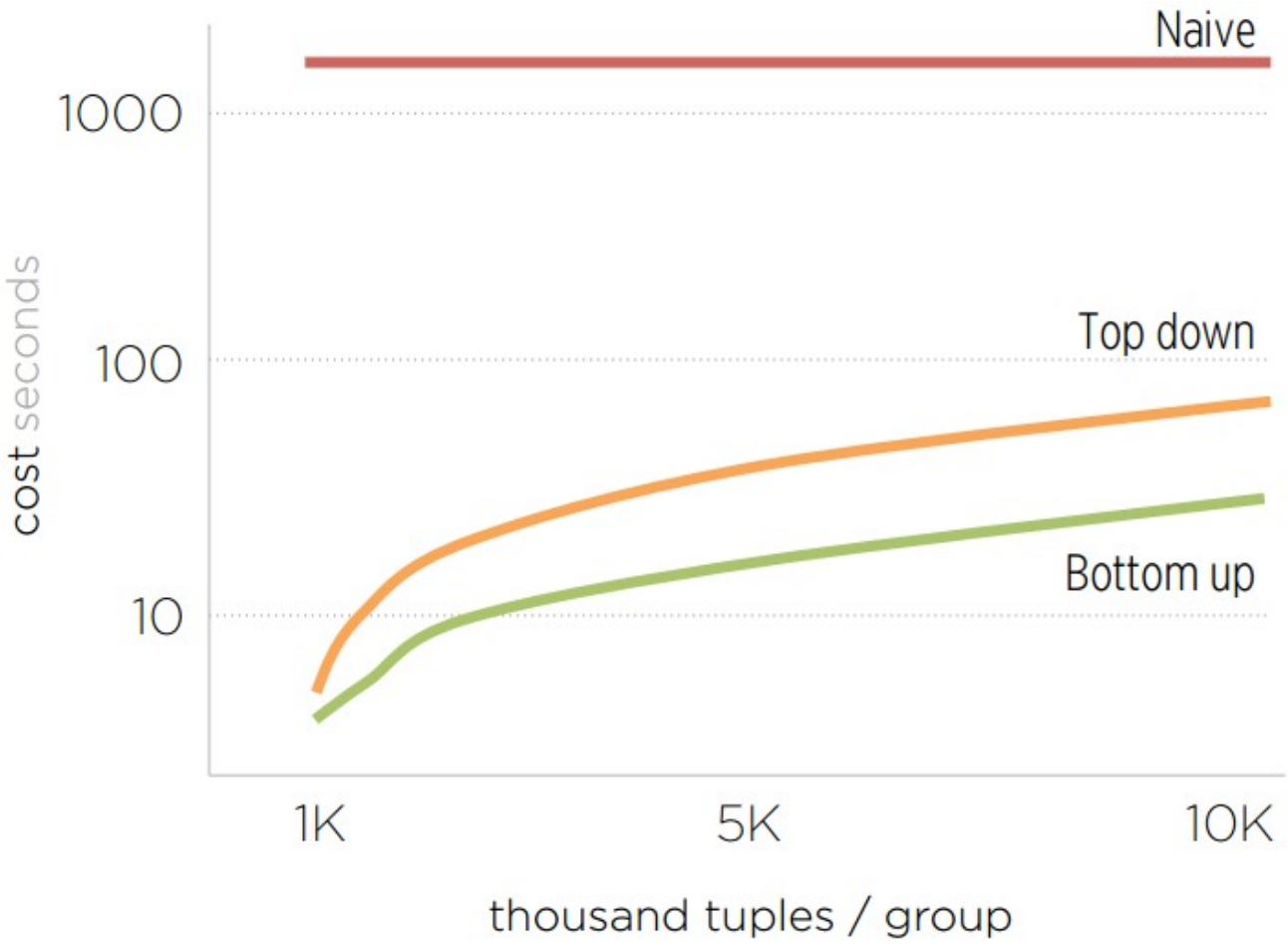
bottom-up:

first search for influential single-attribute predicates, then intersect them to construct multi-attribute predicates.



Properties

- Incrementally removable
- Independent
- Anti-monotonic



Conclusions

- 1st system to explain outliers in aggregation quries
- general system for a large class of why questions
- optimization based on operator properties
  rather than for specific scenario
- clean data automatically and interactively

Thank you!
Q&A