Data Generation using Projection Constraints

Let R be a relation with m tuples and n attributes $\{a_1, a_2, a_3, ..., a_n\}$. You are given count of distinct tuples for every subset of n attributes i.e., you are given $2^n - 1$ constraints of the form:

$$|\pi_{A_i}(R) = k_i| \qquad \text{where } i \in \{1, 2, 3, ..., (2^n - 1)\}$$
$$k_i \in Z^+$$
$$A_i \subseteq \{a_1, a_2, a_3, ..., a_n\}$$

We want to generate data for relation R which satisfies the above constraints. The goals of the project are:

(a) Implement the algorithm presented in [1] and integrate it in the CODD tool [2].

(b) Extend the algorithm implemented in (a) to handle overlapping projection constraints.

(c) Design and implement a novel technique to handle overlapping projection constraints.

(*) A bonus goal is to design and implement a strategy to generate data for relation R which satisfies the above constraints approximately i.e., constraints are met within a δ-threshold.

References

[1] Arvind Arasu, Raghav Kaushik, and Jian Li. Data generation using declarative constraints. SIGMOD, 2011. Sec 4.4.

[2] CODD Metadata Processor. `http://dsl.cds.iisc.ac.in/projects/CODD`, 2015.

Resource Person Raghav Sood

Incremental Algorithm for Solving Filter Constraints

Let R be a relation with m tuples and n attributes $\{a_1, a_2, a_3, ..., a_n\}$.

$$\text{dom}(a_j) = \{1, 2, 3, ..., D\} \qquad a_j \in \{a_1, a_2, a_3, ..., a_n\}$$

We have c filter constraints on R. Each constraint C_i $(1 \leq i \leq c)$ is of the form $< \sigma_i, k_i >$, which means that the number of tuples satisfying the condition σ_i is equal to k_i. σ_i is of the form:

$$\cap \ l_i^j \leq a_j < h_i^j \qquad \text{where } a_j \in \{a_1, a_2, a_3, ..., a_n\}$$

For every tuple $t \in \text{dom}(a_1)$ x $\text{dom}(a_2)$ x ... x $\text{dom}(a_n)$, we create a variablex_t the number of copies of t in R. Now, for each constraint C_i $(1 \leq i \leq c)$, we create a linear equation of the form:

$$\sum_{t:\sigma_i(t)=\text{true}} x_t = k_i$$

This forms a system of c linear equations involving D^n variables. This system can be written in $Ax = b$ format. The goals of this project are:

(a) To implement the algorithm presented in [1] for solving the $Ax = b$ formulation efficiently.

(b) To design and implement an incremental algorithm for the same. Incremental here means that the filter constraints are given in batches and the algorithm should apply next batch of constraints on the current solution in an incremental fashion.

References

[1] Coleman, Thomas F., and Alex Pothen. The Null Space Problem II. Algorithms. SIAM Journal on Algebraic Discrete Methods, 1987.

Resource Person Raghav Sood