Statistical Estimators for Relational Algebra Expressions

Wen-Chu Hou, Gultekin Ozsoyoglu and Baledo K Taneja
Presenter: Alin Dobra

January 26, 2004

The Need for Approximation in Relational Databases

- Relational database technology not directly applicable to *real-time or time-constrained data processing environments*
- Problems with traditional relational databases:
  - cannot be fed real-time data because of monolithic nature
  - software is very large with lots of components: concurrency control, optimization
  - heavily utilizes secondary storage
- Possible solution: *main memory databases*
  - need to fit all data in memory to avoid secondary storage altogether
  - even with all data in memory exact query processing is still expensive

Approximate Query Processing: Sampling Approach

- Select a sample from the database
- Use the sample to construct a *synthetic response* to the requested query
  - construct a *statistical approximation of query responses*
- Focus of the paper:
  - use sampling to determine *consistent and unbiased estimators* for the query results
  - focus only on queries of the form $\mathtt{COUNT}(E)$ with E an relational algebra expression
  - E is allowed to contain $\bowtie$, $\cap$, $\cup$, $-$, $\boldsymbol{\sigma}$ and $\boldsymbol{\pi}$
    - All operators have set semantics (no duplicates)
- No knowledge about the distribution of the data is assumed

Statistical Estimators

Notation:

- Ψ: parameter of interest – population mean
- $\hat{\Psi}$: estimate of parameter Ψ computed from the sample – guess for the real Ψ

Unbiased Estimator:

- $\hat{\Psi}$ is called an unbiased estimator of Ψ if $E[\hat{\Psi}] = \Psi$ for all values of Ψ
- If $\hat{\Psi}$ is not unbiased,

$$\text{bias}(\hat{\Psi}) = E[\hat{\Psi}] - \Psi$$

- The *Mean Square Error* of estimator $\hat{\Psi}$ is defined as:

$$\begin{aligned} \text{MSE}(\hat{\Psi}) &= E((\hat{\Psi} - \Psi)^2) \\ &= \text{Var}(\hat{\Psi}) + (\text{bias}(\hat{\Psi}))^2 \\ &= \text{Var}(\hat{\Psi}) \quad \text{if } \hat{\Psi} \text{ unbiased} \end{aligned}$$

Consistent Estimators: $\hat{\Psi}$ is consistent if $\hat{\Psi} \to \Psi$ when the number of samples goes to infinity (or all tuples in database here)

Statistical Estimators (cont)

- All estimates have error – sample size is finite
  - have to estimate the error of the estimator
  - *confidence bounds*: interval around estimate in which the true value lies with high (prescribed) probability

Determining Confidence Intervals:

- **Idea:** $\hat{\Psi}$ is usually an average or sum of averages
  - Central limit theorem $\Rightarrow$ distribution of $\hat{\Psi}$ is normal
  - if $\hat{\Psi}$ is unbiased and $\mathrm{Var}(\hat{\Psi})$ is known than the confidence interval is

$$E[\hat{\Psi}] \pm z \times \sqrt{\mathrm{Var}(\hat{\Psi})}$$

    where z is the value for $N(0,1)$ that corresponds to the desired confident interval

- If $\hat{\Psi}$ is not normally distributed
  - can use Chebyshev's theorem to give pessimistic, distribution independent bounds

ESTIMATE_COUNT(E) Algorithm

- Input: an arbitrary relational algebra expression E
- Output: an estimate of $\texttt{COUNT}(E)$

1. Push projection inside union. For term E_i

$$\pi\,(\cup_m E_{im}) = \cup_m \pi(E_{im})$$

   $\pi(E_{im})$ considered a relation

2. Transform E into $E_1\phi_1 \ldots \phi_{n-1}E_n$ with $\phi_i \in \{\cup, -\}$ with E_i not containing these type of operators.

3. Compute estimator $\hat{C}_j$ of $\texttt{COUNT}(E) = \sum_j (\pm)\texttt{COUNT}(E'_j)$ using the inclusion exclusion principle.

   For each $\texttt{COUNT}(E'_j)$ chose the appropriate estimator depending if E'_j contains or not π

   Return $\sum_j (\pm)\hat{C}_j$

Example 1: Overall Algorithm

Estimate:

$$\texttt{COUNT}(E) = \texttt{COUNT}(R_1 \bowtie (R_2 - \pi(R_3 \cup R_4 - R_5)))$$

1.
$$R_1 \bowtie (R_2 - \pi(R_3 \cup R_4 - R_5)) = R_1 \bowtie (R_2 - ((\pi(R_3 - R_5)) \cup (\pi(R_4 - R_5))))$$

2. Notation:

$$R_3^* = \pi(R_3 - R_5)$$
$$R_4^* = \pi(R_4 - R_5)$$

$$R_1 \bowtie (R_2 - \pi(R_3 \cup R_4 - R_5)) = (R_1 \bowtie R_2) - ((R_1 \bowtie R_3^*) \cup (R_1 \bowtie R_4^*))$$

3.
$$\begin{aligned}\texttt{COUNT}(E) &= \texttt{COUNT}(R_1 \bowtie R_2) - \texttt{COUNT}((R_1 \bowtie R_2) \cap ((R_1 \bowtie R_3^*) \cup (R_1 \bowtie R_4^*))) \\ &= \texttt{COUNT}(R_1 \bowtie R_2) - \texttt{COUNT}(R_1 \bowtie (R_2 \cap R_3^*)) - \texttt{COUNT}(R_1 \bowtie (R_2 \cap R_4^*)) \\ &\quad + \texttt{COUNT}(R_1 \bowtie (R_2 \cap R_3^* \cap R_4^*)\end{aligned}$$

Estimating $\mathtt{COUNT}(R_1 \bowtie \cdots \bowtie R_n)$

Idea:

- Relation R with k tuples can be mapped to a set of k points in one-dimensional space.
- Crossproducts $R_1 \times \cdots \times R_n$ can be represented as a point in an n-dimensional space with $d_1, \ldots, d_n$ projections in each direction.

  Call the mappings $f_i(\dot{})$
- Natural join and intersection (particular form or natural join) can be represented as a subset of all the possible points in this space.
- Alternative: assign value 1 to each point $p(f_1(t_1), \ldots, f_n(t_n))$ if $(t_1, \ldots, t_n) \in R_1 \bowtie \cdots \bowtie R_n$ and 0 otherwise.

$$\mathtt{COUNT}(E) = \text{number of 1s in the mapping}$$

To solve this subproblem it is enough to estimate the number of 1s.

Estimating number of 1s

Notation:

- N_i number of tuples in R_i
- $N = N_1 \times \cdots \times N_n$
- Assume points are numbered $p_1 \dots p_N$ (any enumeration is fine)
- y_i value 0 or 1 for point p_i
- $Y(E) = y_1 + \cdots + y_N$ is the total number of 1s – $\texttt{COUNT}(E)$

Estimator for $Y(E)$: With S an uniform random sample of points in $R_1 \times \cdots \times R_n$

$$\hat{Y}(E) = N \frac{\sum_{p_i \in S} y_i}{|S|}$$

Can show:

- $\hat{Y}(E)$ is an unbiased estimator of $Y(E)$
- $$\mathrm{Var}(\hat{Y}(E)) = N^2 \frac{N - |S|}{|S|^2} \frac{\sum_{p_i \in S} (y_i - \bar{y})^2}{N-1}, \quad \bar{y} = \frac{\sum_{p_i \in S} y_i}{|S|}$$

Incorporating $\cup, -, \pi$

Operators $\cup$ and $-$: Use inclusion exclusion principle.

This can be achieved by applying transformation rules that push the operators $\cap$ and $-$ outside and then using properties of $\mathtt{COUNT}()$.

$$\mathtt{COUNT}(R_1 - R_2) = \mathtt{COUNT}(R_1) - \mathtt{COUNT}(R_1 \cap R_2)$$

Operator π:

- Difficulty is in eliminating duplicates
- Must not count contribution of a tuple twice
- Goodman estimator: estimates number of distinct values (groups) in a relation

$$\sum_i A_i x_i$$

with x_i the number of sample points with value i and

$$A_i = 1 - (-1)^i \frac{[N - |S| + i - 1]^{(i)}}{|S|^{(i)}}$$

$n^{(i)} = n(n-1)(n-i+1)$ if $i > 0$, 1 otherwise

Sampling For Aggregate Estimation

- Just need uniform samples from crossproduct spaces for the most part
- Can obtain such samples by picking random tuples in each of the participating relations
- Samples can be reused form multiple estimations
- Use nonuniform samples: make all combinations from samples from multiple relations
  - Computation of variance more difficult since the samples are not iid
- Clustered sampling: sample blocks instead of tuples