Approximate Query Processing

Serkan Kumyol 20466705
Abishek Puri 20143602
What is APQ?

SELECT COUNT(*) FROM SALES GROUP BY PRODUCT, STORE, COUNTRY WHERE ..... OR

Results within an error bound in significantly faster time.
Contents

- Measuring Error
- Non-Sampling Based Methods
- Sampling Based Methods
Measuring Error

For a single aggregate query: \( \frac{\text{ExactAnswer} - \text{APQAnswer}}{\text{ExactAnswer}} \)

For a group-by query: Average error for all groups
Basic APQ Strategy

1) Pre-compute some form of summary of the data
   a) Samples, Histograms, Wavelets etc

2) Run query on the summary form

3) Give Error estimate with data
Non-Sampling Based Methods

- Histograms
- Wavelets
Histograms

Key Idea:

1) Partition the rows based on an attribute into buckets

2) Aggregate queries use buckets information
1-D Histograms

Put all the rows into 1-D array of buckets.

<table>
<thead>
<tr>
<th></th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>18</th>
</tr>
</thead>
</table>

Many ways to construct this array of buckets. Same interval size, same number of rows in each bucket, etc.
## Equi-Depth

Same number of rows in each bucket, given $X$ number of buckets.

<table>
<thead>
<tr>
<th>R.A</th>
<th>R.B</th>
<th>R.C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

$X = 3$
Equi-Depth

SELECT COUNT(*) WHERE R.A >= 9 and R.A <= 19

Result = # Rows/Bucket * # Buckets in range = 3.5*#Rows/6

Error : Worst case is that essentially ALL of last bucket is <= 19, so you miss .5*#Rows.
Other 1-D Histogram Implementation

- Minimize Frequency Variance across buckets [Ioannidis, Poosala 95]
  - $O(B*N^2)$ time complexity [Jagadish et al. 98]
Multi-Dimensional Histogram

Key Concept:

- Put rows into corresponding bucket in N-Dimensional Space
- Create a singular point to represent that bucket i.e. approx. location of all points in that bucket
Types

- GenHist [Gunopulos et al 90]
- STHoles [Bruno et al 01]
- [Muralikrishna, DeWitt 88]
- MHIST [Poosala, Ioannidis 97]
Issues

- Curse of Dimensionality kicks in for multi-dimensional
- Requires significant adjustments to the SQL engine
Wavelets

- Mathematical decomposition technique
- Breaks down data into “resolutions”
## Example

<table>
<thead>
<tr>
<th>R.A</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>9</th>
<th>8</th>
<th>10</th>
<th>12</th>
<th>10</th>
</tr>
</thead>
</table>

- \([0, 0, 1, 9, 8, 10, 12, 10]\)
- \([0, 5, 9, 11]\)    \([0, -4, -1, 1]\)
- \([2.5, 10]\)       \([-2.5, -1]\)
- \([6.25]\)          \([-3.75]\)
Sometimes cannot keep ENTIRE error tree in memory, so you must select certain coefficients to discard.

Then error can be computed relative to the number of coefficients discarded. [Matias 98]
Multi-Dimensional Wavelets

- Have been researched, but not deeply
  - [Stollnitz et al 96]
  - [Chakrabarti et al 00]
Sampling Based APQ

• Main claim: Instead of running queries on entire data, a sample $S$ obtained from the same data $R$ is enough to create synopses of data.

  • For a fast approximate answer, apply the query to $S$ & “scale” the result.

  • Intuitively, using a random sample of rows from the R to obtain a sample representing R is expected.

    • But statistics never guarantees that the distribution of the sample is equal to population

      $S \sim R$
Sampling - Basics

• Assume that red circle is a sample represents %20 of the entire dataset then the query:

Select 5*count(*) from S where S.a = 0
result is 5*2 = 10
Sampling - Basics

- *Randomized sampling*: Uniform Random Sampling

- Deterministic(Biased) sampling: Adding bias to the sample which covers outliers and skewness of the dataset.
Sampling – Non-Uniform/Biased Sampling

• Sampling procedure assigns different rates to different data with respect its’ distance to mean.

• Each tuple j is selected for the sample S with some probability $P_j$.

<table>
<thead>
<tr>
<th></th>
<th>R.a</th>
<th>20</th>
<th>20</th>
<th>20</th>
<th>40</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pj</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
<td>1/2</td>
<td>1/2</td>
<td></td>
</tr>
<tr>
<td>Sf</td>
<td>---</td>
<td>3</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>2</td>
</tr>
</tbody>
</table>

Query

Select sum(R.a)
from R
from S
where R.b < 5
Result: sum(R.a) = 140

Sampled APQ

Select sum(S.a * S.sf)
where S.b < 5
Result: 20*3 + 40*2 = 140
Sampling – Stratified Sampling

• Stratified random sampling is a method of sampling that involves the division of a population into smaller groups known as strata. Population -> Stratum -> Strata

• The main advantage of stratified sampling is that it captures key population characteristics in the sample. Furthermore, stratum size can be defined by user. (sample size/population size) x stratum size
Sampling – Stratified Sampling

Common dimensions used for querying and ensures that each dimension or strata have enough representation in the sampled data set.

![Stratified Sampling](https://snappydatainc.github.io/snappydata/sde/key_concepts)

Original Table

<table>
<thead>
<tr>
<th>ID</th>
<th>Advertiser</th>
<th>Geo</th>
<th>Bid</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>adv10</td>
<td>NY</td>
<td>0.0001</td>
</tr>
<tr>
<td>2</td>
<td>adv10</td>
<td>VT</td>
<td>0.0005</td>
</tr>
<tr>
<td>3</td>
<td>adv20</td>
<td>NY</td>
<td>0.0002</td>
</tr>
<tr>
<td>4</td>
<td>adv10</td>
<td>NY</td>
<td>0.0003</td>
</tr>
<tr>
<td>5</td>
<td>adv20</td>
<td>NY</td>
<td>0.0001</td>
</tr>
<tr>
<td>6</td>
<td>adv30</td>
<td>VT</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Stratified Sample on Geo

<table>
<thead>
<tr>
<th>ID</th>
<th>Advertiser</th>
<th>Geo</th>
<th>Bid</th>
<th>Sampling Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>adv20</td>
<td>NY</td>
<td>0.0002</td>
<td>1/4</td>
</tr>
<tr>
<td>2</td>
<td>adv10</td>
<td>VT</td>
<td>0.0005</td>
<td>1/2</td>
</tr>
</tbody>
</table>

```
SELECT avg(bid)
FROM AdImpressions
WHERE geo = 'VT'
```
Sampling – Error Estimation

- There are several ways to measure errors in sampling some of them are:
  - Confidence Intervals (CI) [Acharya et al. 2000, Agarval et al. 2013]
  - Mean Squared Error [Chaudhuri et al. 2001, Babcock et al. 2003]
  - Bootstrap [Zeng et al. 2014]
Sampling Architectures

• Two types of architectures:
  • Query time sampling
    • Online aggregation
    • No a priori knowledge about the data, user is responsible of choosing sampling operations in SQL queries.
  • Pre-computed sampling
    • Query independent sample(s)
    • Synopses are very helpful
Pre-computed Sampling Architecture

Pre-processing

Database → Build Samples

Offline

Workload

Online

Rewrite and Execute

Incoming Query

Answer set with error estimate

Query Processing
Sampling – Uniform Sampling

• Sampling procedure assigns different rates to different data in order.

Bernoulli Sampling
- Expected sample size is $f*N$
  where $f$ is the sampling rate

Reservoir Sampling
- Fixed sample size $f*N$
  - Each new item accepted with appropriate probability.
  - If sample is full, delete a random item from sample
  - ItemSkip

Figure: https://snappydatainc.github.io/snappydata/sde/key_concepts
Uniform Sampling – Trade-off

• Advantages
  • Simple
  • Efficient pre-processing

• Disadvantages
  • Performs low on sparse data
  • Insensitive to variance
  • Low selectivity queries
Uniform Sampling – Limitations for Aggregation Queries

- Skewed data in aggregate values and large variance.

<table>
<thead>
<tr>
<th>ProductId</th>
<th>Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>1000</td>
</tr>
</tbody>
</table>

aggregate query  Q: SELECT SUM(Revenue) FROM R
R: 1030

Sample data with rate of %50

Two different uniform samples gives us wrong answers when we run aggregate query.

S1 = {<1, 10>, <3, 10>} = 40
S2 = {<1,10>, <4,1000>} = 2020
Uniform Sampling – Limitations for Aggregation Queries

• Low selectivity:
  • Some queries involve conditions and group-by
  • If selectivity of the query is low, it is highly possible for uniform sample set to fall out of target. When we sample sub-relations it is also sampled with sample rate

• The effect of small groups is same as effect of low selectivity
Solution: Implementing Various Sampling Methods.

- Outlier Indexing [Chaudhuri et al. 2001]
- ICICLES (Self-Tuning) [Ganti et al. 2000]
- Congressional Sampling [Acharya et al. 2000]
- AQUA Project [Acharya et al. 2000]
- Strat [Chaudhuri, Das, Narasayya 2001]
- Dynamic Sample Selection [Babcock, Chadhouri, das 2003]
  - Small group sampling
- Sampling over multi-table databases using F-K Joins. [Acharya et al. 1999]
Biased Sampling: Outlier Indexing

• Basic idea:
  • Treat outliers separately from the data
  • Use uniform sampling for rest of the relations
  • Use uniform sample and outlier index together to answer queries
Biased Sampling: Outlier Indexing

• Algorithm works as:
  • Sort the values
  • Find the mean and detect values with highest variance add them to outlier index

• Shortcomings:
  • Constant update of samples is required.
  • Not useful for Count, Max, Min but useful for Sum, Avg
  • Global outliers vs. Local outliers
Biased Sampling: ICICLES

• An answer to low selectivity and small groups problem.
• Based on the exploiting workload information.
• Weighted sampling of data by exploiting workload information while drawing the sample.
  1. Workload collection: $w = \{q_1, q_2, \ldots\}$
  2. Trace query patterns
  3. Trace tuple usage
  4. Perform sampling according to weights obtained from frequently queried data points.
Biased Sampling: ICICLES

• Given:
  • Relation R
  • Workload W: $Q_1, ..., Q_n$
  • Limited sample space

• Find and maintain a sample $S_w(R)$
Biased Sampling: ICICLES

• R(Q): set of tuples in R
• Where each tuple $t_i$ has weight $w_i$

$$p_i = \frac{n \cdot w_i}{\sum_{j=1}^{N} w_j}$$

• Then ICICLE $S_w(R)$ is a random sample of $R \cup^+ R(Q_1) \cup^+ ... \cup^+ R(Q_1)$
Biased Sampling: ICICLES - Issues

• ICICLES has to be maintained.
  • Update $S_w(R)$ for new query $Q$: $L \cup^+ R(Q)$
  • Usually Reservoir Sampling is used

• Restricting the workload: which queries will be included to samples. Overhead vs. Precision
  • Restrict query with respect to representational power of sample and overhead.

• No variance handling
• No claims about optimality
• But guaranteed quality on sampled queries
• Problem with group-by queries – Solution Congressional Sampling
Biased Sampling: Congressional Sampling

- Useful for approximating GroupBy queries.
  - Congressional samples are a hybrid of uniform and biased samples.
  - Decision support queries segment data into groups.
  - Congress Algorithm: Considers a “workload” of all subsets of grouping columns, and attempts to design a sample such that all groups get equal importance.
Biased Sampling: Congressional Sampling

• At the end, we obtain a stratified, biased sample in which each group which is a partitioning of it’s own strata.

• Since we have samples from each stratum, both small and large groups have sample to answer groupby queries.

• Shortcomings:
  • Allocation of samples is not necessarily optimal
  • Data variance is not considered
Biased Sampling: STRAT

• A unified approach: Outlier Indexing, ICICLES and Congressional Sampling.

• Based on stratified sampling

• «Given a workload of queries, we select a stratified random sample of the original data such that the error in answering the workload queries using the sample is minimized.»[Chaudhuri, Das, Narassaya 2001]
Biased Sampling: STRAT

Diagram showing regions $R_1$, $R_2$, $R_3$, $R_4$, $R_{Q1}$, $R_{Q2}$, and $R_{Q3}$.
Biased Sampling: STRAT

• How to generalize query workload to rest of the sample?

- Lifting the workload information
  Deriving query distribution from given workload.
  - \( P(Q') > P(Q'') \)
  - \( P(Q|W) \)
Biased Sampling: STRAT

• Steps in STRAT:
  • Stratification
    • Find strata in R: \{R_1, ..., R_r\}
  • Allocation
    • Total of K unknown samples.
    • Stratum R_j is used for deriving a new sample K_j where error for lifted workload is minimized.
  • Obtained distribution of lifted query calculated according to Mean Squared Error, \text{MSE}(P_Q) between strata R and workload.
Dynamic Sample Selection

• Previous examples use a fixed biased sample and relatively small disk space because of increasing sample size increases running time of the query against sample. [Babcock, Chaudhuri, Das, 2003]

• Dynamic Sampling:
  - Pre-processing: Creating a large sample containing a family of differently biased sub samples
  - Run-time: Using only related portion to answer the query
Dynamic Sample Selection

- Data distribution and query distribution used for identifying set of biased samples
- Samples are created and stored in database together with meta data.

Figure: Pre-Processing
Dynamic Sample Selection

- Rewrite query.
- Choose samples for given query according to metadata of the samples.

Figure: Run-time phase
Dynamic Sample Selection: Small Group Sampling

• Intuition: Treat small and large groups differently.
  • Distributional properties of small and large groups are different.

Large groups uniform random sample
Small groups use original data
Dynamic Sample Selection: Small Group Sampling

• Finding small groups:
  • Look for rare values in columns
  • Identify rare values during pre-processing
  • Create small groups tables from rows
  • Query time: scan small group table for each grouping attribute
Dynamic Sample Selection: Small Group Sampling

- Run query on small group table for each grouping attribute
- Run scaled query on all samples
Dynamic Sample Selection

• An accurate model with cost of disk space.
• Solves biased sampling approach’s distribution problem.
• Selects best sample for incoming query.

• Small Group Sampling
  • Store large and small groups differently.
  • Small groups are cheap to scan.
  • Advantageous compared to Congressional sampling
Problem with Joins

• Until now, we looked at base sampling methods mostly uniform sampling combined with certain biases.

• There are two issues about joins:
  • Join of two uniform random samples is not necessarily uniform.
  • Small join result size.

• Pre-computing the joint samples looks like inevitable.
  • Computing join samples by executing all possible join queries.
Non-Sampling Summary

- Histograms
  - 1-D
  - Multi-Dimensional

- Wavelets
  - Haar 1-D Wavelets
  - Multi-Dimensional Wavelets
Sampling Summary

- Uniform Sampling
- Biased Sampling
  - ICICLES
  - Outlier Indexing
  - Congressional Samples
  - STRAT
- Dynamic Sampling
References


5. Gautam Das: *Survey of Approximate Query Processing Techniques*. (Invited Tutorial) SSDBM 2003


8. GANTI, Venkatesh; LEE, Mong-Li; RAMAKRISHNAN, Raghu. *Icicles: Self-tuning samples for approximate query answering*. In: VLDB. 2000