



TOWARDS INTELLIGENT DATA PROFILING AND AGGREGATION



Nidhi Menon
Sneha Venkatachalam



AGENDA

- INTRODUCTION
- RELATED WORK
- KEY IDEAS
- MOTIVATION
- IMPLEMENTATION
- PERFORMANCE MEASURE
- QUESTIONS & DISCUSSIONS

INTRODUCTION

- Data Profiling
 - Useful in Data pre-processing and analytics
 - Summarize data
- Data Aggregation
 - Useful in Data Science and Statistics
 - Eliminate repetitive Calculation of Statistics
- Key Idea: Intelligent data profiling and aggregation for faster query retrieval

RELATED WORK

Data Canopy: Accelerating Exploratory Statistical Analysis

A. Wasay, X. Wei, N. Dayan, and S. Idreos, "Data Canopy: Accelerating Exploratory Statistical Analysis," in ACM SIGMOD International Conference on Management of Data, 2017

Profiling relational data: a survey

Abedjan, Ziawasch, Lukasz Golab, and Felix Naumann. "Profiling Relational Data: A Survey." The VLDB Journal 24.4 (2015): 557–581

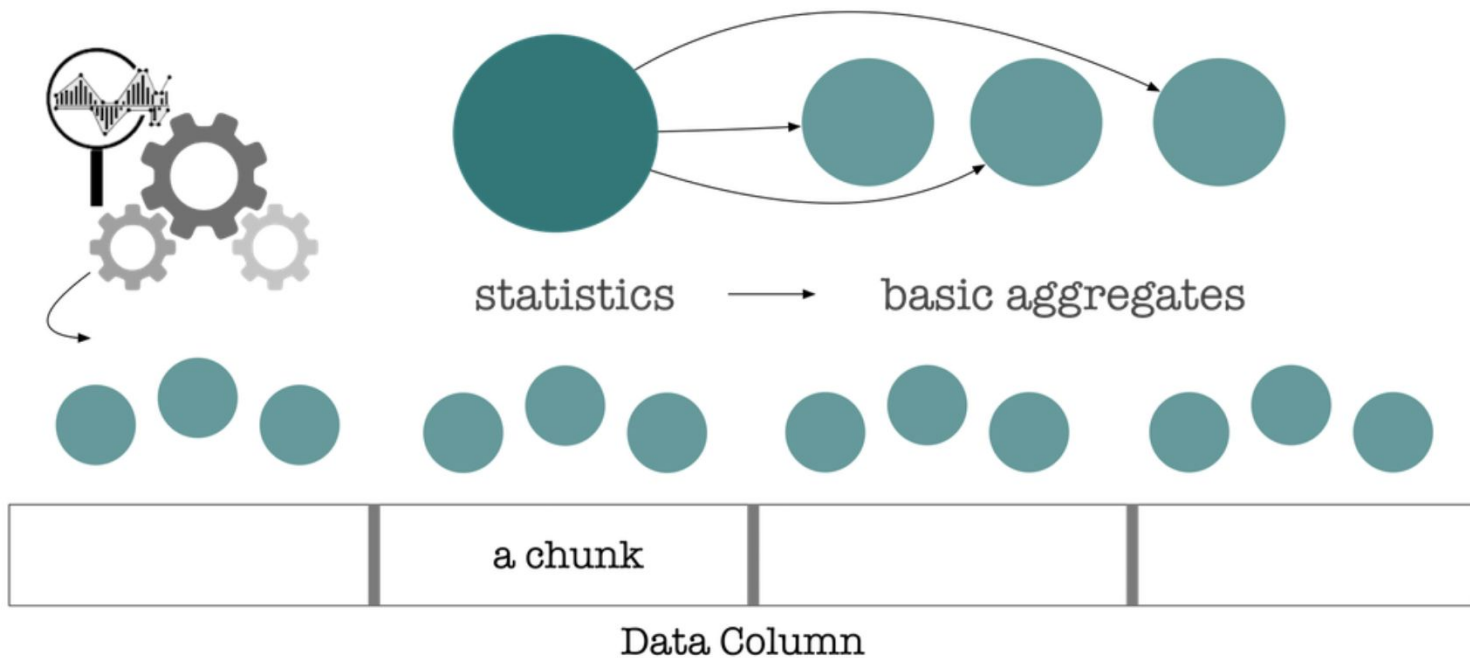
DATA PROFILING

- Task of reviewing data from an existing source to understand its structure, content and relationships
- Aids us in computing statistics or in collective informative summaries about the data
- Data Profiling tasks include:
 - **Single-column tasks**
 - **Multi-column tasks**
 - Dependency detection
- A set of results of these tasks gives a ***data profile*** or ***database profile***

Category	Task	Description
Cardinalities	num-rows	Number of rows
	value length	Measurements of value lengths (minimum, maximum, median, and average)
	null values	Number or percentage of null values
	distinct	Number of distinct values; sometimes called “cardinality”
Value distributions	uniqueness	Number of distinct values divided by the number of rows
	histogram	Frequency histograms (equi-width, equi-depth, etc.)
	constancy	Frequency of most frequent value divided by number of rows
	quartiles	Three points that divide the (numeric) values into four equal groups
	first digit	Distribution of first digit in numeric values; to check Benford’s law
Patterns, data types, and domains	basic type	Generic data type, such as numeric, alphabetic, alphanumeric, date, time
	data type	Concrete DBMS-specific data type, such as varchar, timestamp.
	size	Maximum number of digits in numeric values
	decimals	Maximum number of decimals in numeric values
	patterns	Histogram of value patterns (Aa9...)
	data class	Semantic, generic data type, such as code, indicator, text, date/time, quantity, identifier
	domain	Classification of semantic domain, such as credit card, first name, city, phenotype

[Profiling relational data: a survey]: Overview of selected single column profiling tasks

STATISTICAL CALCULATIONS



REPETITIVE STATISTICS

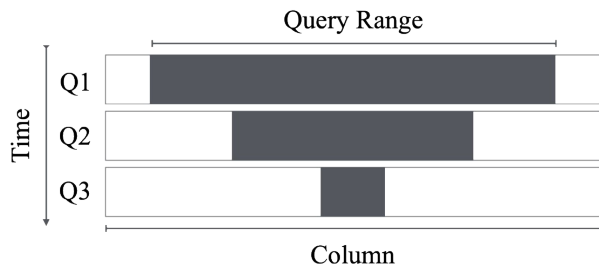


Fig.: Sub-range

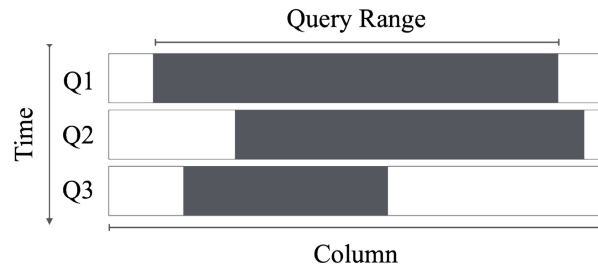


Fig.: Overlap

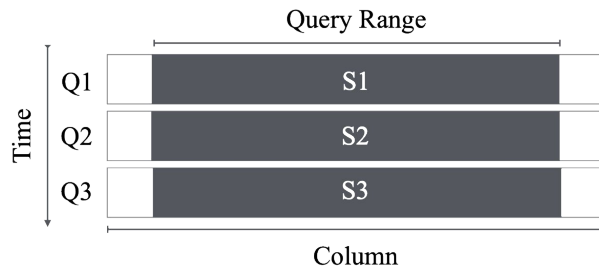


Fig.: Different Statistics

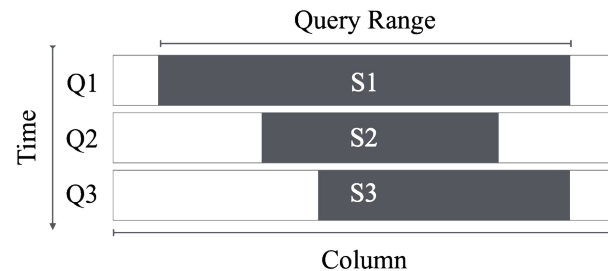
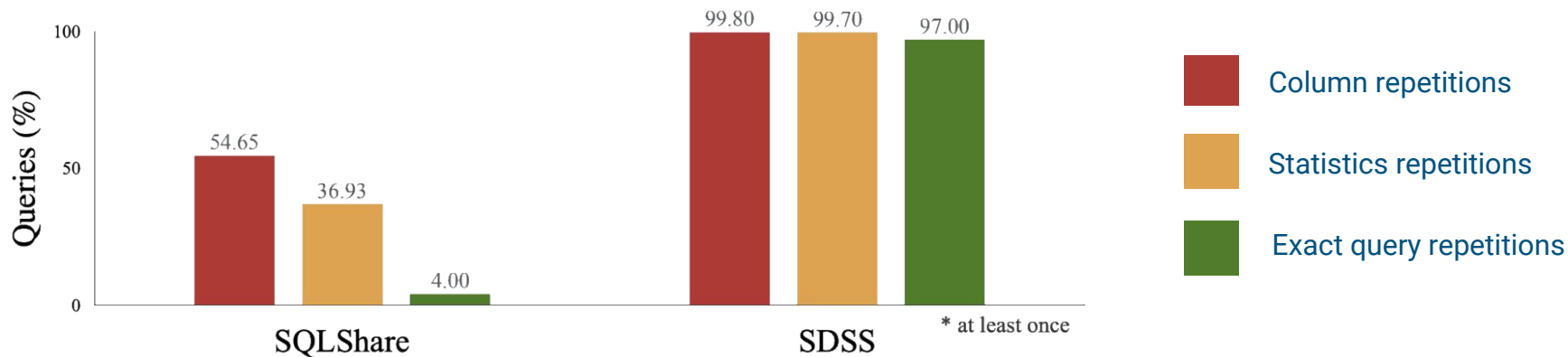


Fig.: Mixed

MOTIVATION

Exploratory Workloads Exhibit Repetition

- Repetition is everywhere - between 50% to 99%



SQLShare: Results from a Multi-Year SQL-as-a-Service Experiment

Shrainik Jain, Dominik Moritz, Bill Howe, Ed Lazowska. SIGMOD 2016

IMPLEMENTATION

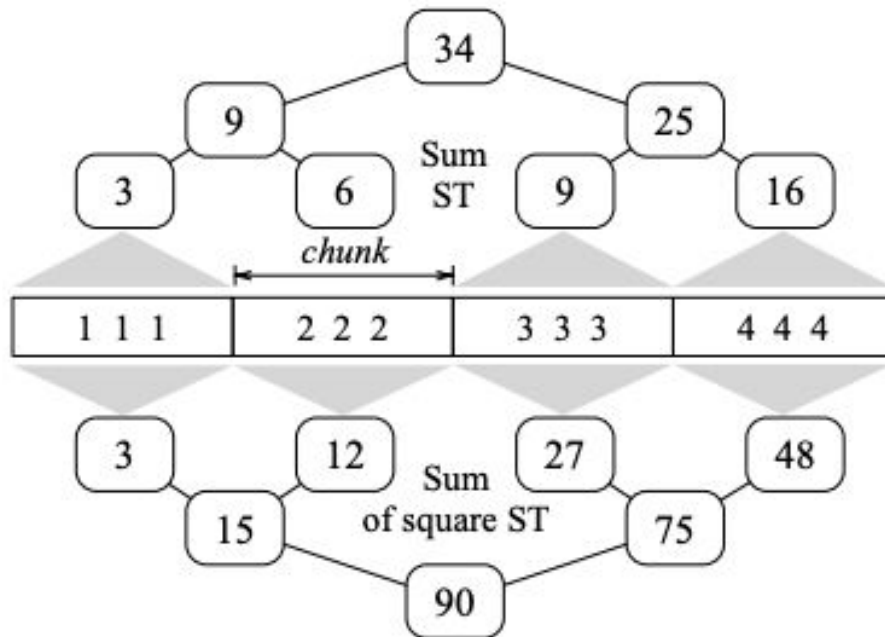
- **Language:** C++
- **Dataset (Numerical):** Randomly generated; uniform distribution; ~10k rows
- **Query structure:** queryMethod (low, high, column_name)

SEGMENT TREE

A tree data structure that stores data about intervals, or segments

- Binary tree
- Leaves: data instances
- Internal nodes: union of elementary intervals

SEGMENT TREE



PROGRESS UPDATE

- Data Aggregation
 - Segment tree implementation for caching
 - Build segment tree for entire dataset
 - Handles querying over continuous ranges
 - Handles updates to the data
 - Hash table implementation for mapping
 - Maps incoming query to corresponding segment tree
 - Statistics computation
 - Statistics mentioned in the 'Data Canopy' paper and 'Data Profiling' paper

Statistics		Basic Aggregates				
Type	Formula	$\sum x$	$\sum x^2$	$\sum xy$	$\sum y^2$	$\sum y$
Mean (avg)	$\frac{\sum x_i}{n}$					
Root Mean Square (rms)	$\sqrt{\frac{1}{n} \cdot \sum x^2}$					
Variance (var)	$\frac{\sum x_i^2 - n \cdot \text{avg}(x)^2}{n}$					
Standard Deviation (std)	$\sqrt{\frac{\sum x_i^2 - n \cdot \text{avg}(x)^2}{n}}$					
Sample Covariance (cov)	$\frac{\sum x_i \cdot y_i}{n} - \frac{\sum x_i \cdot \sum y_i}{n^2}$					
Simple Linear Regression (slr)	$\frac{\text{cov}(x,y)}{\text{var}(x)}, \text{avg}(x), \text{avg}(y)$					
Sample Correlation (corr)	$\frac{n \cdot \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{\sqrt{n \cdot \sum x_i^2 - (\sum x_i)^2} \sqrt{n \cdot \sum y_i^2 - (\sum y_i)^2}}$					

Table of statistics from the paper titled 'Data Canopy' by Wasay et. al.

PROFILING RELATIONAL DATA

- Single column profiling (Category: Cardinalities)
 - **Null values:** Number or percentage of null values

6.3	NULL	19.5	42.7	23.1	NULL	0.0	19.5	35.8	6.3
-----	------	------	------	------	------	-----	------	------	-----



0	1	0	0	0	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---

PROFILING RELATIONAL DATA

- Single column profiling (Category: Cardinalities)

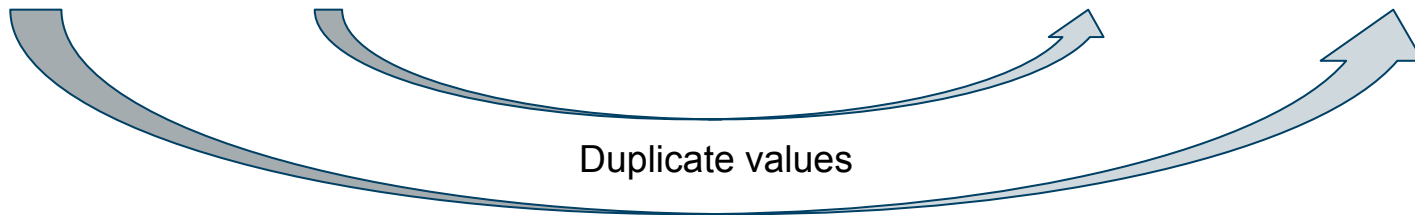
- **Distinct:** Number of distinct values; sometimes called “cardinality”

6.3	NULL	19.5	42.7	23.1	NULL	0.0	19.5	35.8	6.3
-----	------	------	------	------	------	-----	------	------	-----



Hashset to identify duplicates

1	0	1	1	1	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---



PROFILING RELATIONAL DATA

- Single column profiling (Category: Cardinalities)

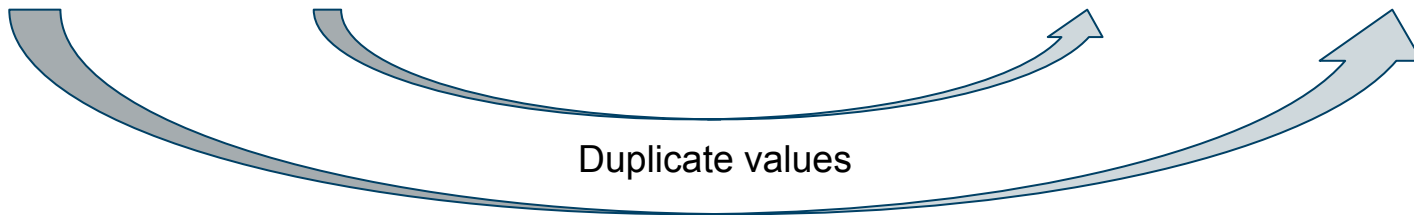
- **Uniqueness**: Number of distinct values divided by the number of rows

6.3	NULL	19.5	42.7	23.1	NULL	0.0	19.5	35.8	6.3
-----	------	------	------	------	------	-----	------	------	-----



Hashset to identify duplicates

1	0	1	1	1	0	1	0	1	0
---	---	---	---	---	---	---	---	---	---



PROFILING RELATIONAL DATA

- Single column profiling (Category: Value Distribution)
 - **Equal-width histogram:** Aggregates for base width 'w' and multiples of 'w'
 - Supported for data types *int* and *float*
 - Based on the concept of binning over base width 'w'

6.3	NULL	19.5	42.7	23.1	NULL	0.0	19.5	35.8	6.3
-----	------	------	------	------	------	-----	------	------	-----



For Width = 5


1	2	0	2	1	0	0	1	1	0
---	---	---	---	---	---	---	---	---	---

0 5 10 15 20 25 30 35 40 45 50

PROFILING RELATIONAL DATA

- Alternate method: count
 - **Equal-width histogram:** Aggregates over any bin-size i.e. width 'w'
 - Supported for data type *int* only
 - Based on the concept of inverted index

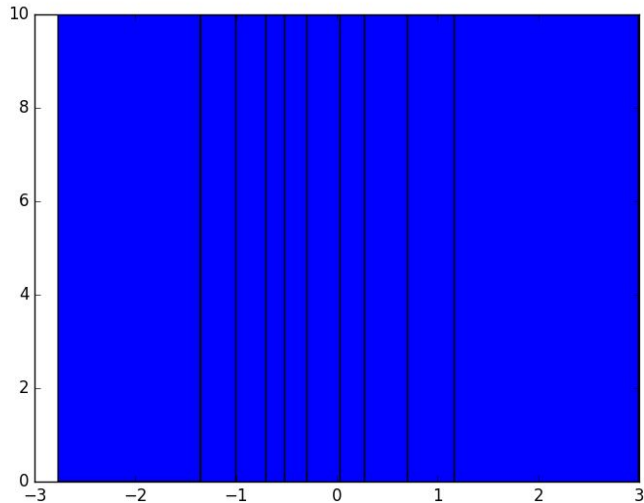
6	NULL	9	4	2	NULL	0	7	3	5
---	------	---	---	---	------	---	---	---	---



1	0	1	1	1	1	1	1	0	1
0	1	2	3	4	5	6	7	8	9

PROFILING RELATIONAL DATA

- Current work
 - Equal-height histogram
 - To handle querying of equal count of values over dynamic-sized ranges, and updates



PERFORMANCE MEASURE

- Time Complexity
 - Tree Construction: $O(n)$
 - $2(n)$ nodes, value of each node calculated once in tree construction
 - Tree Query: $O(\log n)$
 - Number of levels: $O(\log n)$
 - To query a range minimum, at most 2 nodes at every level processed
- Space Complexity
 - Tree: $O(n)$
 - For n data instances, segment tree uses a $2(n)$ sized array

PERFORMANCE MEASURE

- MEMORY: Comparison of traditional implementation vs. our implementation
 - **Traditional**
 - No memory overhead assuming dynamic calculations for all statistics
 - **Our Implementation**
 - For int/float values (10k data instances)
 $10,000 * 2 * 4 = 80,000 \text{ bytes} = 80 \text{ KB}$

PERFORMANCE MEASURE

- **SPEEDUP:** Comparison of our implementation vs. traditional implementation

```
Mean: 5002.88
Time taken by function: 39 microseconds

RMSE: 5774.25
Time taken by function: 11 microseconds

Variance: 3.3342e+07
Time taken by function: 8 microseconds

STD: 5774.25
Time taken by function: 6 microseconds

Covariance: 86945.9
Time taken by function: 16 microseconds

Simple Linear Regression: 0.0026675
Time taken by function: 16 microseconds

Correlation: 0.0104589
Time taken by function: 6 microseconds
```

```
Mean: 5002.88
Time taken by function: 75 microseconds

RMSE: 5774.25
Time taken by function: 399 microseconds

Variance: 3.33419e+07
Time taken by function: 344 microseconds

STD: 5774.25
Time taken by function: 308 microseconds

Covariance: 86916.4
Time taken by function: 93 microseconds

Simple Linear Regression: 0.00266659
Time taken by function: 438 microseconds

Correlation: 0.0104553
Time taken by function: 725 microseconds
```

QUESTIONS & DISCUSSIONS

