Scalable Visualization and Analytics for Spatial Data

Prof. Chen Li

VLDB Summer School, July 2019, Beijing, China
Outline

1. Motivation
2. Basic concepts
3. Techniques:
   a. Technique: view materialization
   b. Technique: Progressive computation (DRUM)
   c. Technique: VAS
   d. Technique: Sample+Seek
   e. Technique: Marviq
   f. Other studies
4. Cloudberry: an open source approach
Motivation

1. Importance of spatial data
2. Importance of visualization
Importance of spatial data
Importance of visualization

- Determine relationships
- Understand and describe locations and events
- Detect and quantify patterns
- Make predictions
- Find best locations and paths
Earthquakes from 6/9/19 to 7/9/19

Pacific Area

California
5+ year of drought (United States)
Disaster Risk Monitoring and Identification (Cambodia)
Crime rates
Visualization: scatterplot
Visualization: heatmap
Visualization: choropleth
Demo: TwitterMap system
Focus of this tutorial

1. Efficient visualization of large amounts of spatial objects
2. Could have selection conditions, e.g.:
   a. Taxi pickup events on Feb. 1, 2019
   b. Tweets related to “flood”
3. Cover recent studies in this research area
Multi-tier architecture
Cloudberry: Middleware for Big Data Visualization
Outline

1. Motivation
2. Basic concepts
3. Techniques:
   a. Technique: view materialization
   b. Technique: Progressive computation (DRUM)
   c. Technique: VAS
   d. Technique: Sample+Seek
   e. Technique: Marviq
   f. Other studies
4. Cloudberry: an open source approach
2. Basic concepts

1. Spatial data
2. Data storage (heap file)
3. Indexing structures: B+ tree, R-tree
4. Query Plans
Spatial data types

1. Point
   - (x, y) or (longitude, latitude)

1. Line Segment
   - [(x₁, y₁), (x₂, y₂)]

1. Polygon
   - [(x₁, y₁), (x₂, y₂), …, (xₙ, yₙ)]

1. Trajectory
   - [(x₁, y₁, t₁), (x₂, y₂, t₂), …, (xₙ, yₙ, tₙ)]
Storing records in files

Heap file
B+ tree
Hierarchical index (e.g., R-tree)
R-tree
Heatmap
Heatmap SQL query

```sql
SELECT width_bucket(coordinate[0], -123.257481, -114.137924, 180) as bx,
width_bucket(coordinate[1], 31.979682, 41.967915, 200) as by,
COUNT(*)
FROM twitter
WHERE create_at between '2017-06-01 00:00:00' and '2017-06-30 24:00:00'
AND coordinate <@ box '((-123.257481, 31.979682), (-114.137924, 41.967915))' //
within California and Nevada
GROUP BY bx, by;
```
Physical Plans

- B+Tree on create_at
- RTree on coordinate
- B+Tree on (create_at, x, y)
Outline

1. Motivation
2. Basic concepts
3. Techniques:
   a. Technique: view materialization
   b. Technique: Progressive computation (DRUM)
   c. Technique: VAS
   d. Technique: Sample+Seek
   e. Technique: Marviq
   f. Other studies
4. Cloudberry: an open source approach
Technique: Query acceleration using view materialization

Towards Interactive Analytics and Visualization on One Billion Tweets, Jianfeng Jia, Chen Li, Xi Zhang, Chen Li, Michael J. Carey, Simon Su, ACM SIGSPATIAL 2016 (Demo Paper)

Main idea:
- A user submits a query to visualize tweets of “Zika”
- “Zika” tweets are materialized as a table
- Future queries related to Zika can be answered using this table
View caching and incremental computation

1. Suggest View "drug"
2. Create New Dataset "drug"
3. Write MetaData ("drug", May 18)
4. Update View periodically
What if no views available?
**Technique:** Progressive computation (DRUM)

Drum: A Rhythmic Approach to Interactive Analytics on Large Data, Jianfeng Jia, Chen Li, Michael J. Carey, IEEE Big Data 2017

Main idea:
- Deliver results progressively
- Maintain a fixed rhythm
Fixed-length slicing?
Query slicing with a rhythm
What is a good slicing schedule?
Drum: Adaptive Framework for Query Slicing
Schedule cost

- Total running time
- Smoothness of result delivery

\[ \text{Cost}(S) = \text{Cost}_t(S) + \alpha \sum D_i. \]
Linear regression with uncertainty
Tradeoff of Running Time and Penalty
Choosing $r_i$ to maximize the expected score
Drum: Summary

- Slice a query to small queries based on a selection attribute
- Goal:
  - Reduce initial response time
  - Deliver results progressively
  - Optimize the “rhythm”
- Other attributes for slicing
  - E.g., spatial slicing
Technique: Visualization-Aware Sampling (VAS)

Yongjoo Park, Michael J. Cafarella, Barzan Mozafari: Visualization-aware sampling for very large databases. ICDE 2016: 755-766

1. Motivation
2. Formal definition of good sample
3. Approximation algorithm
Motivation: scatterplot on many spatial points

100M points

<table>
<thead>
<tr>
<th>Software</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>matplotlib</td>
<td>71 mins</td>
</tr>
<tr>
<td>MathGL</td>
<td>2+ hours</td>
</tr>
<tr>
<td>Tableau</td>
<td>X</td>
</tr>
</tbody>
</table>
We want:
- **Reduce** computational effort
- **Without** affecting visual perception
Idea: sampling

Question 1: What is a **good** sample?
Question 2: **How to** obtain it?

\(|S| < |D| \rightarrow \text{faster}\)

Data Reduction
Different sampling methods

- Original (2 billion points)
- Uniform random sample (1 million)
- Stratified sample (1 million)
- VAS (1 million)
Good for 3 common visualization-driven tasks

1. Trend Analysis
2. Density Est.
3. Clustering
Technique: VAS

1. Motivation
2. Formal definition of good sample
3. Approximation algorithm
Quality of sample

What is a **good** sample (S) of the original dataset (D)?

$$\text{Loss}(S) = \text{D} - \text{S}$$
What is a good sample \((S)\) of the original dataset \((D)\)?

\[
\text{subloss}(x) = D - S
\]
What is a **good** sample (S) of the original dataset (D)?

\[
\text{subloss}(x) = D - S
\]
What is a **good** sample \((S)\) of the original dataset \((D)\)?
What is a **good** sample \((S)\) of the original dataset \((D)\)?

\[
\text{subloss}(x) = \frac{\|D - S\|_2}{\|D\|_2}
\]

If the visual difference for **every circle** is small,

\[\rightarrow\text{two viz will look similar}\]
What is a *good* sample (S) of the original dataset (D)?

\[ \text{subloss}(x) = \]  

<table>
<thead>
<tr>
<th>Goal</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>To minimize:</td>
<td></td>
</tr>
<tr>
<td>[ \text{Loss}(S) = \int \text{subloss}(x) dx ]</td>
<td></td>
</tr>
<tr>
<td>where subloss(x) is the viz distance for the circle centered at x.</td>
<td></td>
</tr>
</tbody>
</table>
What is a good sample \( (S) \) of the original dataset \( (D) \)?

\[
\text{subloss}(x) = \quad \quad D \quad \quad \quad \quad S
\]

<table>
<thead>
<tr>
<th>Goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>To minimize:</td>
</tr>
<tr>
<td>[ \text{What function to use for subloss}(x) \ ? ]</td>
</tr>
<tr>
<td>[ \text{Loss}(S) = \int \text{subloss}(x) dx ]</td>
</tr>
<tr>
<td>where subloss( (x) ) is the viz distance for the circle centered at ( x ).</td>
</tr>
</tbody>
</table>
Let’s see what happens \textit{around} the location \( x \).

Initially, visual distance around \( x \) reduces fast

Visual distance round \( x \) \textit{no longer} reduces much
Let's see what happens around the location x.

\[ \text{subloss}(x) = \frac{1}{\sum_{s_i \in S} K(x, s_i)} \]

where $K$ captures the proximity between two coordinates.
**VAS: problem formulation**

Given a budget $|S| = K$, we want

$$\arg\min_{S} \int_{S} \text{subloss}(x)dx$$

subject to $S \subseteq D \land |S| = K$

$$\int \frac{1}{\sum_{s_i \in S} K(x, s_i)} dx$$
Given a budget $|S| = K$, we want

$$\arg \min_S \int_{\text{subloss}(x)} dx \quad \text{s.t. } S \subseteq D \land |S| = K$$

$$\int \frac{1}{\sum_{s_i \in S} K(x, s_i)} dx \quad \text{NP-hard}$$
Technique: VAS

1. Motivation
2. Formal definition of good sample
3. Approximation algorithm
Approximation Algorithm

Nemhauser, et al. [2] algorithm (comes with an error guarantee)
As scanning over a dataset,

Valid replacement: $\text{Loss}(S') < \text{Loss}(S)$
Testing for valid replacement is too slow: $O(K^3)$ for every point
Approximation Algorithm

Expand/Shrink operation

Expand/Shrink is fast: $O(K)$ for every point $\rightarrow O(K^2)$ times faster!

The result is exactly the same as Nemhauser's algorithm.
User Study

Smaller Loss(S) $\rightarrow$ More successes
User Study

Average user performance

1. Uniform Random Sampling
2. Stratified Sampling
3. Visualization-Aware Sampling (VAS)

success: if a user answers a question correctly.
Performance

VAS Offline Sample Time vs Sample Ratio

Sample Ratio
- 5%
- 10%
- 15%

Sample Time (Seconds)
VAS: Summary

- Sampling for visualization
- Proposed a quality function
- Proposed an algorithm to compute a sample
VAS: observations

- Too slow for online arbitrary query visualization
- Users must give a K (sample size), hard to choose
- Loss of original dataset Loss(D) ≠ 0; counterintuitive
- Not a similarity function:
  - better to allow user to input some error threshold (error < 1%)
Technique: Sample+Seek


1. Motivation
2. Distribution Precision
3. Architecture
4. Solution
5. Experiments
SELECT state, SUM(sales) 
FROM T 
GROUP BY state 

{CA: 190} 
{NY: 298}

CA:39% 
NY:61%
We want an error metric to capture the similarity of distributions!
Technique: Sample+Seek

1. Motivation
2. Distribution Precision
3. Architecture
4. Solution
5. Experiments
\[ \sqrt{\sum_{\text{group}[i]} \left( \frac{\text{group}[i]'s \ value}{\text{total group value}} - \frac{\text{estimated group}[i]'s \ value}{\text{total est. group value}} \right)^2} \]

\[ = \sqrt{(0.39 - 0.66)^2 + (0.61 - 0.34)^2} \]
Technique: Sample+Seek

1. Motivation
2. Distribution Precision
3. Architecture
4. Solution
5. Experiments
**T**

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>198</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>

**SELECT state, SUM(sales)**

**FROM T**

**GROUP BY state**

**Exact Answer (f)**

- CA: 39%
- NY: 61%

- \|f^* - f|_2 | x | ε

**Approximate Answer (f^*)**

- CA: 40%
- NY: 60%
Technique: Sample+Seek

1. Motivation
2. Distribution Precision
3. System Overview
4. Solution
5. Experiments
### Table T

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>198</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>

### Uniform Sample

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
</tbody>
</table>

### Offline

- **SAMPLE SIZE**: \( \frac{1}{\epsilon^2} \)
- **CA**: 50%
- **NY**: 50%

### Online

- **SELECT** state, \( \text{COUNT}(*) \)
- **FROM** T
- **GROUP BY** state

- **CA**: 50%
- **NY**: 50%

- **Sample Size \( \geq \frac{1}{\epsilon^2} \)**

- **w.h.p., we have** \( \|f^* - f\|_2 \leq \epsilon \)**
### Uniform Sample

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>198</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>

### Online

With a Predicate

\[
\text{SELECT state, COUNT(\ast) FROM T WHERE T.model = 'Benz' GROUP BY state}
\]

\[
\begin{align*}
(\text{CA}) & : 9 \\
(\text{NY}) & : 10
\end{align*}
\]

Result: \(\text{Size} \geq \frac{1}{\epsilon^2}\)

w.h.p., we have \(\|f^\ast - f\|_2 \leq \epsilon\)
### Measure-biased Sample

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>92</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>98</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>198</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>

**Online**

```sql
SELECT state, SUM(sales) FROM T GROUP BY state
```

**Offline**

- **CA**: 39%
- **NY**: 61%

w.h.p., we have $||f^* - f||_2 \leq \epsilon$

Sample Size $\geq 1/\epsilon^2$
### T

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>101</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>198</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>

### Offline Measure-biased Sample

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>92</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>98</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
</tbody>
</table>

### Online

\[
\text{SUM with Predicate}
\]

\[
\begin{align*}
\text{SELECT } & \text{state, SUM(sales)} \\
\text{FROM } & T \\
\text{WHERE } & \text{T.model = 'Benz'} \\
\text{GROUP BY } & \text{state}
\end{align*}
\]

\[
\text{COUNT(*)}
\]

### Measure-biased Sample

- CA: 40%
- NY: 60%

### Offline

- CA: 39%
- NY: 61%

w.h.p., we have \[\|f^\hat{\cdot}-f\|_2 \leq \epsilon\]

### Result

Size \(\geq 1/\epsilon^2\)

Same Generalization
**Seek:** when not enough sample

For low-selectivity predicates:

```sql
SELECT state, SUM(*)
FROM T
WHERE T.model = 'Audi'
GROUP BY state
```

**Result** \( \text{Size} < 1/\epsilon^2 \)

### Measure-biased Sample

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>92</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>98</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>1</td>
</tr>
<tr>
<td>199</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>w</td>
<td>100</td>
</tr>
<tr>
<td>200</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
<tr>
<td>...</td>
<td>NY</td>
<td>Benz</td>
<td>b</td>
<td>100</td>
</tr>
</tbody>
</table>
**Seek:** when not enough sample

For low-selectivity predicates:

```
SELECT state, SUM(*)
FROM T
WHERE T.model = 'Audi'
GROUP BY state
```

<table>
<thead>
<tr>
<th>ID</th>
<th>state</th>
<th>model</th>
<th>color</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>91</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>CA</td>
<td>Audi</td>
<td>w</td>
<td>10</td>
</tr>
</tbody>
</table>

Kind of view materialization...
Seek: What if conjunction has low selectivity?

Low-Frequency Group Index can not help

```sql
SELECT state, SUM(*)
FROM T
WHERE T.model = 'Benz'
AND T.color = 'w'
GROUP BY state
```
Seek: What if conjunction has low selectivity?

```sql
SELECT state, SUM(*)
FROM T
WHERE T.model = 'Benz'
AND T.color = 'w'
GROUP BY state
```
### Seek: What if conjunction has low selectivity?

- **Measure Augmented Index**
  - **model**
    - Benz
    - Audi
  - **color**
    - w
    - b

### Table

<table>
<thead>
<tr>
<th>ID</th>
<th>sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>1</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td>1</td>
</tr>
<tr>
<td>200</td>
<td>100</td>
</tr>
<tr>
<td>ID</td>
<td>sales</td>
</tr>
<tr>
<td>----</td>
<td>-------</td>
</tr>
<tr>
<td>91</td>
<td>10</td>
</tr>
<tr>
<td>...</td>
<td>10</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
</tr>
</tbody>
</table>

- **Intersect**

- **Online**
  - **Measure-biased Sampling**
    - ID | state | model | color | sales |
    - 1  | CA    | Benz  | w     | 1     |
    - ...| CA    | Benz  | w     | 1     |
    - 90 | CA    | Benz  | w     | 1     |
    - 91 | CA    | Audi  | w     | 10    |
    - ...| CA    | Audi  | w     | 10    |
    - 100| CA    | Audi  | w     | 10    |
    - 101| NY    | Benz  | w     | 1     |
    - ...| NY    | Benz  | w     | 1     |
    - 198| NY    | Benz  | w     | 1     |
    - 199| NY    | Benz  | w     | 100   |
    - 200| NY    | Benz  | b     | 100   |

### T

- **ID** | **state** | **model** | **color** | **sales** |
- 1      | CA        | Benz      | w         | 1         |
- ...    | CA        | Benz      | w         | 1         |
- 90     | CA        | Benz      | w         | 1         |
- 91     | CA        | Audi      | w         | 10        |
- ...    | CA        | Audi      | w         | 10        |
- 100    | CA        | Audi      | w         | 10        |
- 101    | NY        | Benz      | w         | 1         |
- ...    | NY        | Benz      | w         | 1         |
- 198    | NY        | Benz      | w         | 1         |
- 199    | NY        | Benz      | w         | 100       |
- 200    | NY        | Benz      | b         | 100       |
Technique: Sample+Seek

1. Motivation
2. Distribution Precision
3. Architecture
4. Solution
5. Experiments
- TPC-H schema (with eight tables - 300M rows in LINEITEM)
- Queries with 1-4 group-by dimensions, and 0-4 predicate dimensions

100× speedup for > 80% queries

Actual error is “always” smaller than the requested $\epsilon$
(consistent to the theoretical results)
- A real enterprise log table with 30 dimensions and 1B+ rows
- Queries with 1-4 group-by dimensions, and 0-4 predicate dimensions

Sample part scales sub-linearly
Index part tends to be constant

**SPS**: Our method
**DBX**: A commercial RDBMS with columnstore
**BLK**: BlinkDB
**SMG**: SmallGroup sampling

Adjust $\epsilon$
Better accuracy-time tradeoff
Use Sample+Seek for Spatial Viz

- Scatterplot

```sql
SELECT x, y
FROM twitter
WHERE create_at between '2017-06-05 20:00:00'
  and '2017-06-08 08:00:00';
```

- Heatmap

```sql
SELECT width_bucket(x, -127, -63, 480) as bx,
width_bucket(y, 20, 60, 270) as by,
COUNT(*)
FROM twitter
WHERE create_at between '2017-06-05 20:00:00'
  and '2017-06-08 08:00:00'
GROUP BY bx, by;
```
Sample+Seek: limitations

- **No support of progressive computation:** It utilizes a fixed sized sample to approximate queries, so that it can **NOT** keep improving the precision of results progressively.

- **Need to change the DB engine**

- **Distribution Precision (DP) is very insensitive to small groups**

\[
\begin{align*}
\text{X} & : A=55\%, B=44\%, C=1\% \\
\text{Y} & : A=56\%, B=44\%, C=0\% \\
\text{Z} & : A=54\%, B=44\%, C=2\%
\end{align*}
\]

\[\text{DP}(X, Y) = \text{DP}(X, Z)\]

But C is missing in Y.
Technique: Marviq


1. Motivation
2. Visualization Quality
3. MVS and MVS+
4. Construction, Storage and Maintenance
5. Generalization
6. Evaluation
Approximate viz

Tweets containing “fortnite”
Approximate viz

Taxi pickup events in NYC on 1/1/2016
Main idea
System Architecture
Visualization quality

Scatterplot and Jaccard

\[ J(V_1, V_2) = \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|} \]
Quality of approximate viz

\[ J(V_1, V_2) = \frac{|V_1 \cap V_2|}{|V_1 \cup V_2|} = \frac{|V_2|}{|V_1|} \]
MVS: Materialized visualization structure

Exact Visualization (EV)

Scatterplot of records in May, 2010

\[ \mathcal{J}(V(Q), V_\alpha) \geq \frac{|V_\alpha|}{|V_\theta|} \]
MVS

Alpha to Theta:

\[ |V_\alpha| = 37 \]
\[ |V_Q| = 39 \]
\[ |V_\theta| = 4 \]

\[ \mathcal{J}(V(Q), V_\alpha) \geq \frac{|V_\alpha|}{|V_\theta|} = \frac{37}{44} \]
Storing MVS as tables in DB

Parent_Interval

<table>
<thead>
<tr>
<th>Parent_ID</th>
<th>Start</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Parent_Pixel

<table>
<thead>
<tr>
<th>Parent_ID</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/2010</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5/2010</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>5/2010</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MVS construction

1. Given intervals
2. Given workload & storage constraint
3. Adaptive construction
Incremental Maintenance

EV can be maintained incrementally for inserts and deletes:
Improvement: MVS+

Low-resolution Visualization (LV)

AEV\_{[10/2018, 12/2018]} \rightarrow ALV_{10/201}
Improvement: MVS+

$\text{MVS: quality}=\frac{|a|}{|\Theta|}$, retrieving records in $\gamma$. 

Query range $C$

2013 to 2016

2017
Improvement: MVS+

**MVS+:** $\text{quality} = \frac{|a \cup \beta|}{|a \cup \gamma|}$, retrieving $\beta$.

**Benefit:**

1. Reducing the number of retrieved records: $\gamma \rightarrow \beta$
2. Tightening bound: $a \rightarrow a \cup \beta \uparrow$, $\theta \rightarrow a \cup \gamma \downarrow$
MVS+ stored as tables
Multi-attribute Conditions

MVS construction  Quality estimation
Multi-attribute Conditions

MVS construction

Quality estimation
Other quality functions

\[ MSE(A, B) = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} (A(i, j) - B(i, j))^2 \]

\[ MSE(V(Q), V_\alpha) \leq \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} (V_\alpha(i, j) - V_\theta(i, j))^2 \]
Other visualization methods

EV construction for Heatmap

$MSE(V_\theta, V_\alpha) \geq MSE(V(Q(T)), V_\alpha)$
Storage space

Dataset: NYC Taxi, interval size: 3-15 days.
**MVS vs. MVS+**

\[ m = \frac{\text{parent interval length of MVS}^+}{\text{parent interval length of MVS}} \]

NYC Taxi
Other related studies
Kyrix (MIT)

- Declarative model
- On top of DB
- Caching
- Incremental fetching

Kyrix: Interactive Pan/Zoom Visualizations at Scale. Eurographics Conference on Visualization (EuroVis) 2019
Kyrix: Interactive Visual Data Exploration at Scale. Conference on Innovative Data Systems Research (CIDR) 2019
**Kyrix (MIT)**

**Difference**

- **NO arbitrary queries**
  - E.g. time within '2017-06-05 20:00:00' and '2017-06-08 08:00:00'

- **Scalability**
  - Many points in one viewport is Kyrix's future work

- **Kyrix focuses on end-to-end solution**
  - Minimum coding → whole application including frontend
**HadoopViz (UC Riverside)**

**Mapper:**
- Multilevel pyramid partitioning
- Replicate a point to overlapping tiles in each level

**Reducer:**
- Plot an image for each tile
- Images do not need to be merged

HadoopViz: A MapReduce framework for extensible visualization of big spatial data. ICDE 2016: 601-612
HadoopViz (UC Riverside)

Focus
- **Offline**: Using MapReduce Framework to calculate the big multi-layer image tiles efficiently offline
- **Online**: Applying Google Maps API to Zoom and Pan on the multi-layer image tiles

Limitation
- **NOT** support online arbitrary queries
Spatial Online Sampling and Aggregation

- Spatial Online Sampling and Aggregation. PVLDB 2015, Lu Wang, Robert Christensen, Feifei Li, Ke Yi
- RS-Tree:
  - Add a buffer of samples of points from subtree rooted at the node $u$.
  - Each $p \in P(u)$ has equal probability of being an entry in the sample buffer at $u$
Spatial Online Sampling and Aggregation
Spatial Online Sampling and Aggregation

- Start at the root node.
  - Scan the sample buffer, report those within $Q$
  - Descend the tree, only add nodes which intersect $Q$
  - Obtain samples from nodes with appropriate probability during sampling
Spatial Online Sampling and Aggregation

We do not know how many points inside a node satisfy the query

- We sample each node with probability proportional to number of total items in node.
Spatial Online Sampling and Aggregation

Focus
- Online progressively return random samples
- Arbitrary spatial range queries

Difference
- Probabilistic approach
- Modify the DB
  - Hard to support complex queries
    - GROUP BY, Conjunction Predicates
We want to thank Qiushi Bai and Liming Dong who contributed to these slides.

Chen Li
Professor @UCI
https://chenli.ics.uci.edu

Qiushi Bai
PhD student @UCI
https://qiushibai.wordpress.com

Liming Dong
PhD student @Tsinghua University