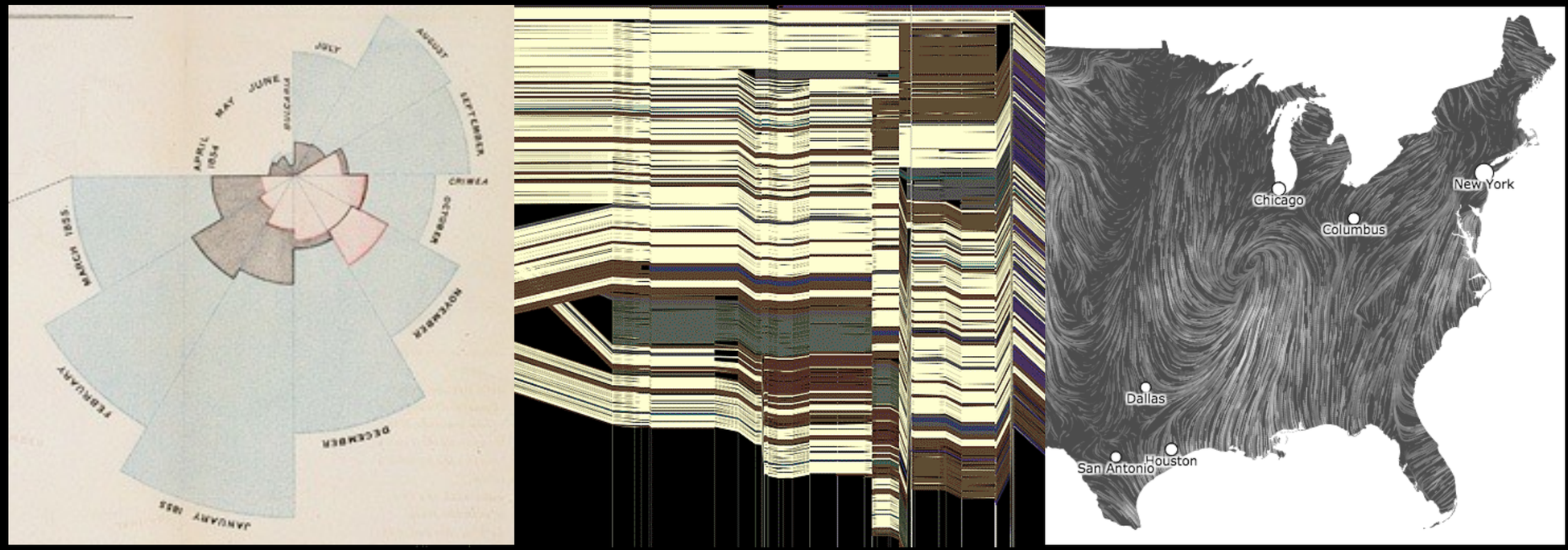
CSE 412 - Data Visualization

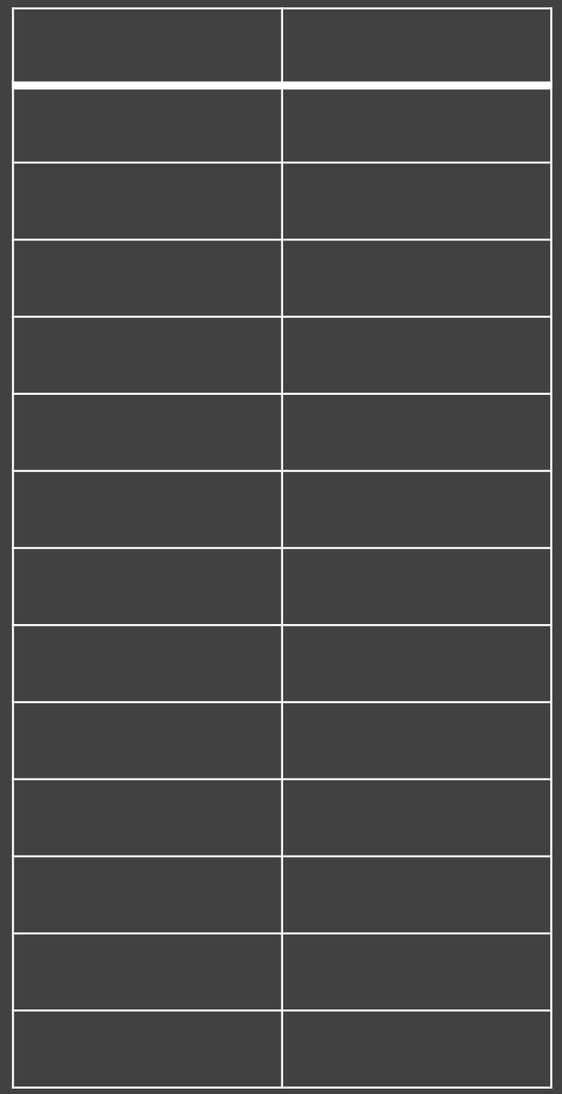
Scalable Visualization



Jeffrey Heer University of Washington

Varieties of “big data”...

Tall Data



Lots of records

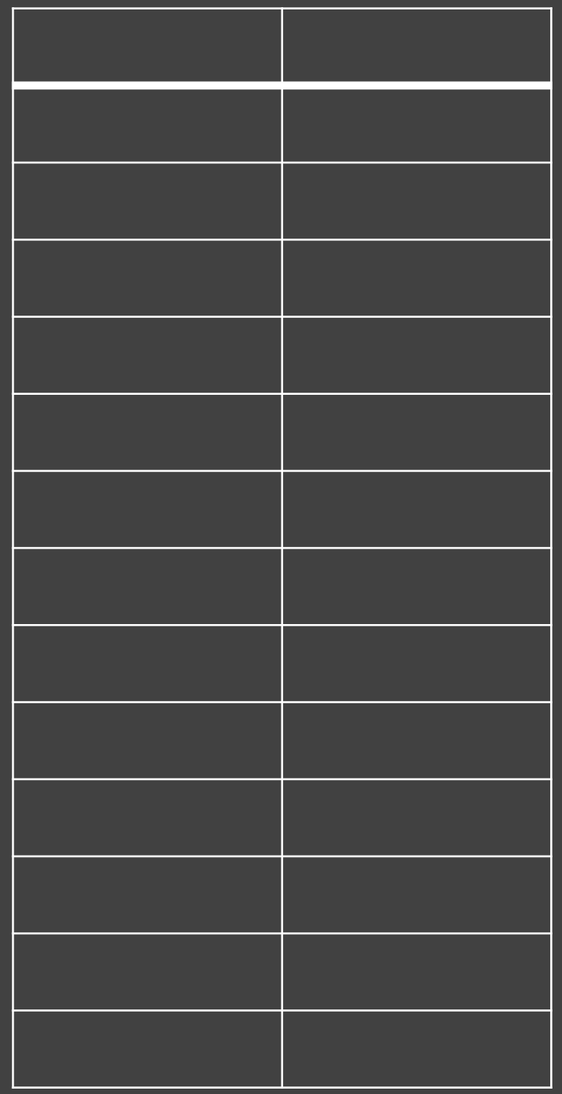
- Large DBs have petabytes or more
- *(but median DB still fits in RAM!)*

How to manage?

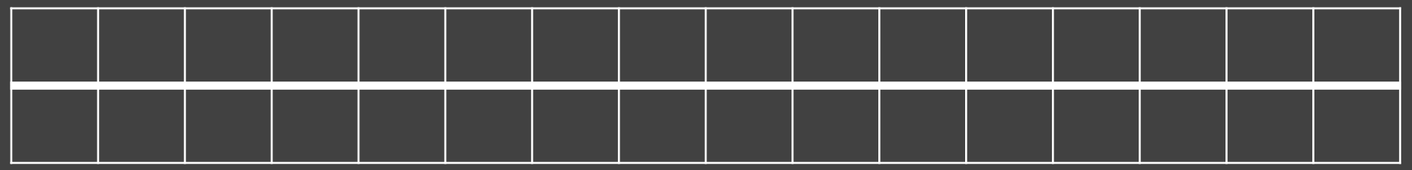
- Parallel data processing
- Reduction: Filter, aggregate
- Sample or approximate

Not just about systems. Consider perceptual / cognitive scalability.

Tall Data



Wide data



Lots of variables (100s-1000s…)

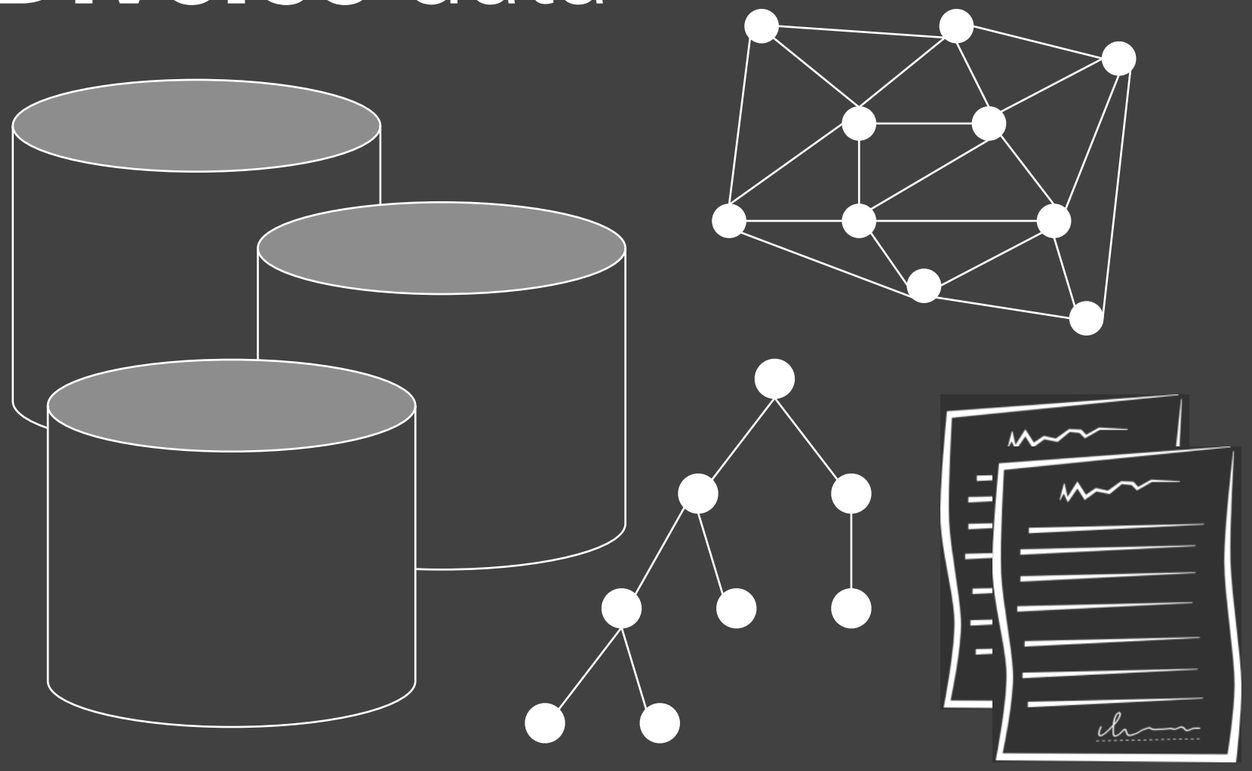
- Select relevant subset
- Dimensionality reduction
- Statistical methods can suggest and order related variables

Requires human judgment

Tall Data

Wide data

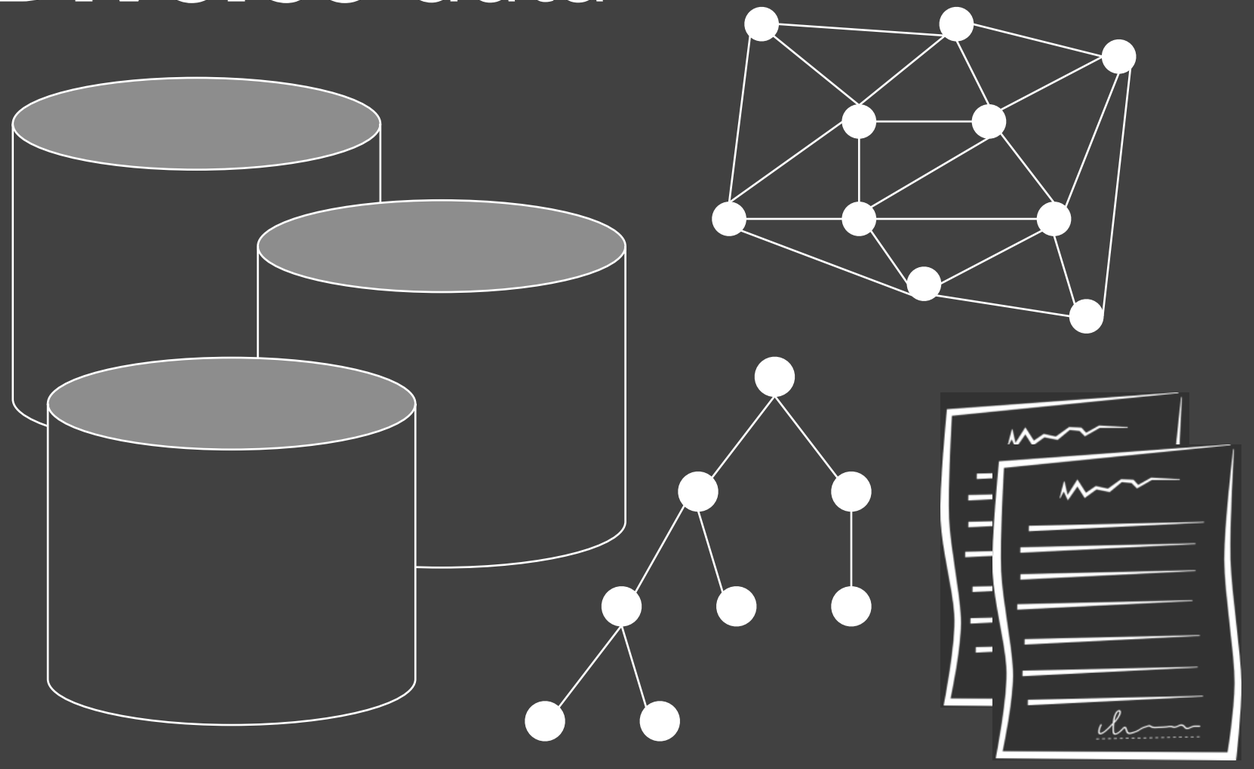
Diverse data



Tall Data

Wide data

Diverse data



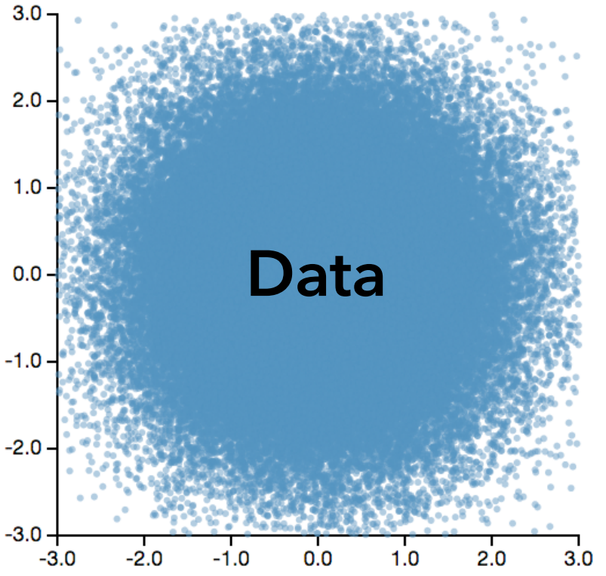
How can we visualize and interact with **billion+ record** databases in real-time?

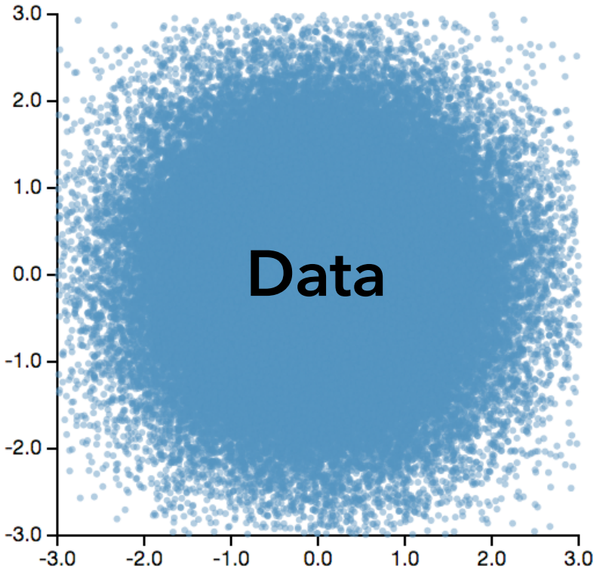
Two Challenges:

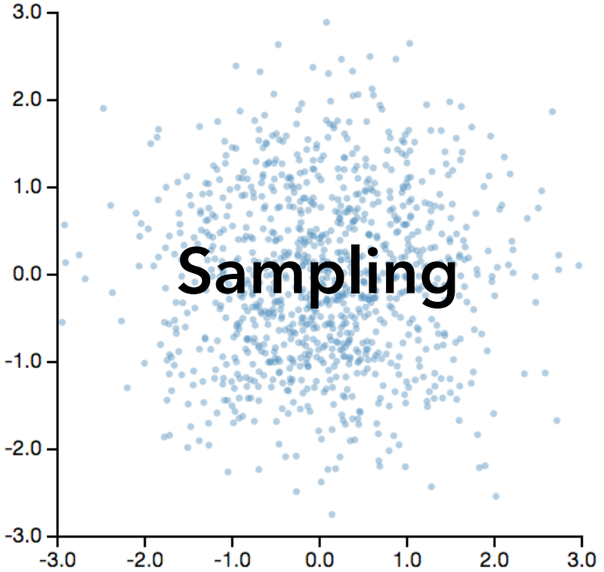
1. Effective **visual encoding**
2. Real-time **interaction**

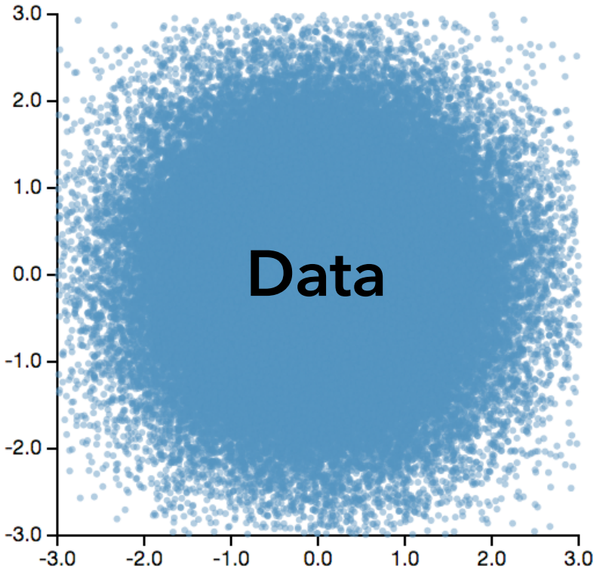
Perceptual and interactive scalability should be limited by the **chosen resolution** of the visualized data, not the number of records.

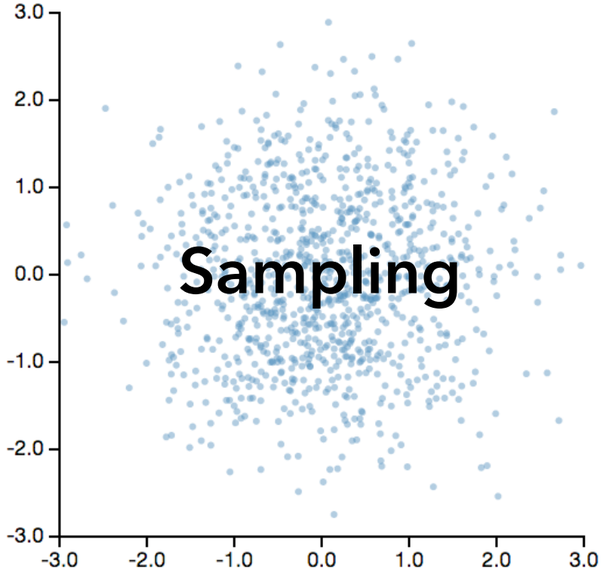
1. Visualizing Large Datasets

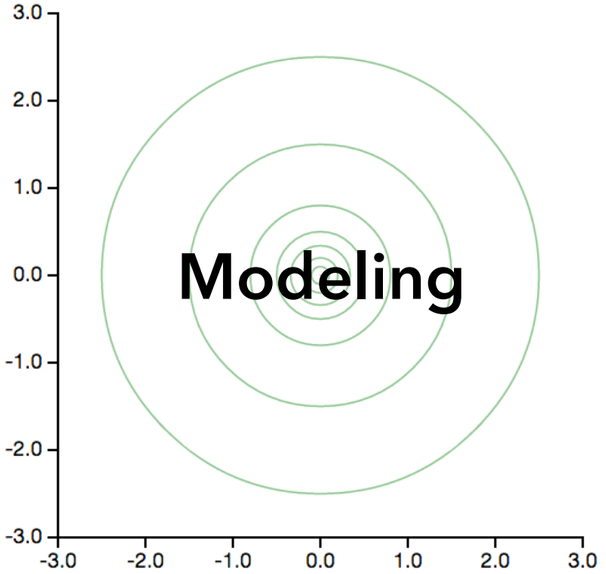

Data
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

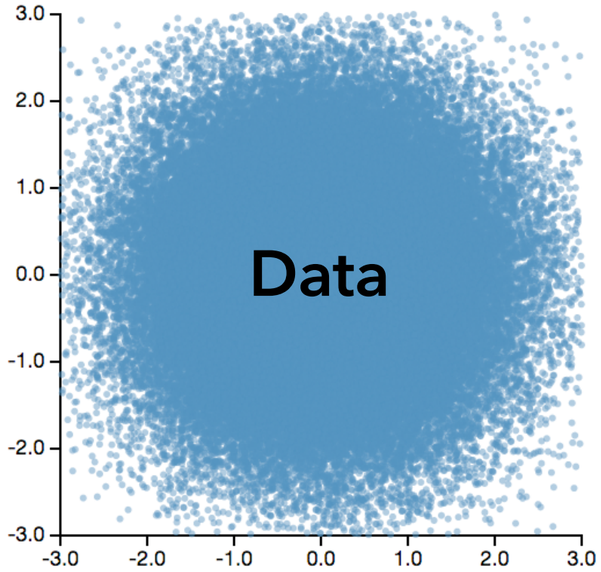

Data
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

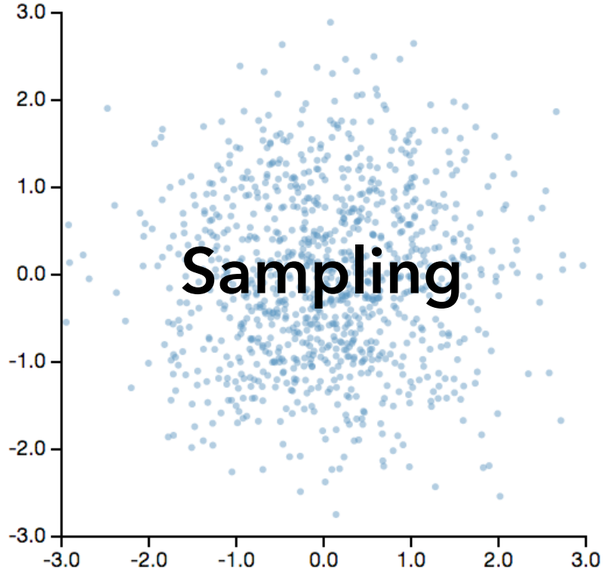

Sampling
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

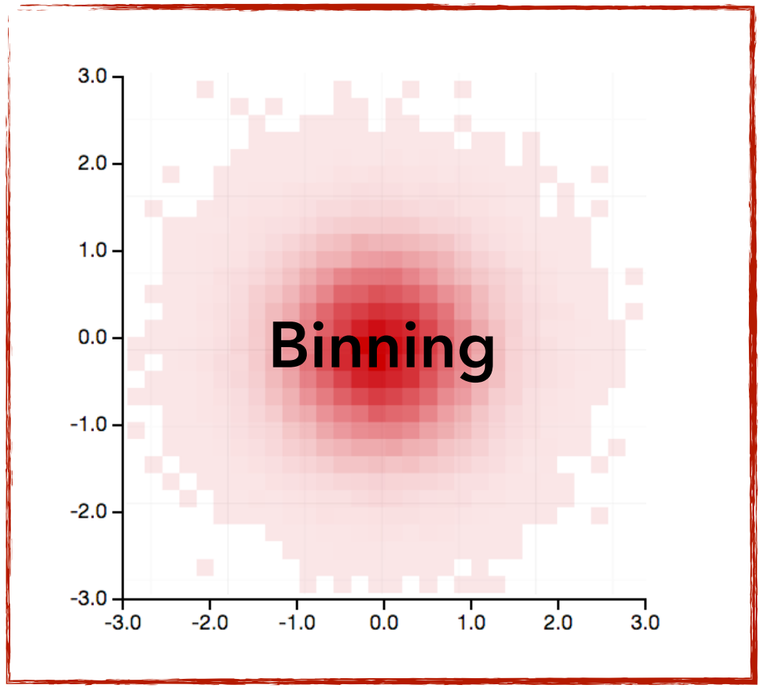

Data
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

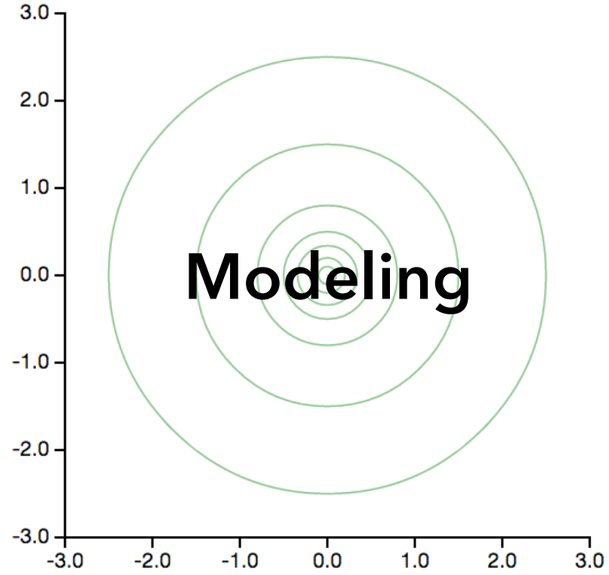

Sampling
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0


Modeling
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

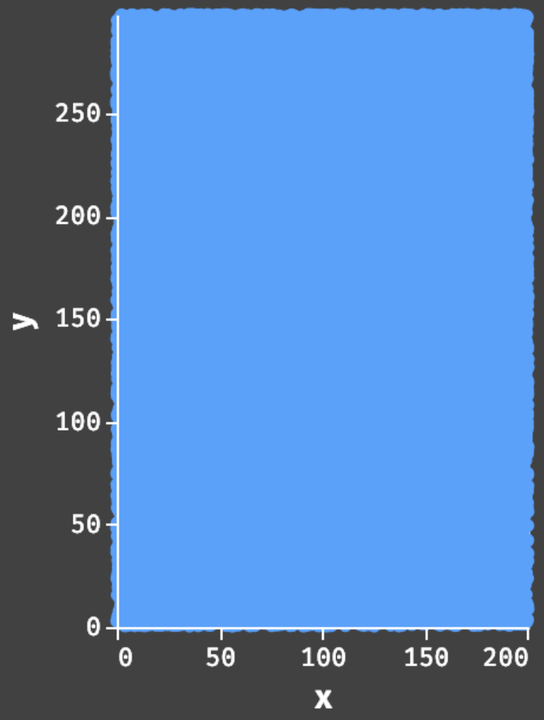

Data
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0


Sampling
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

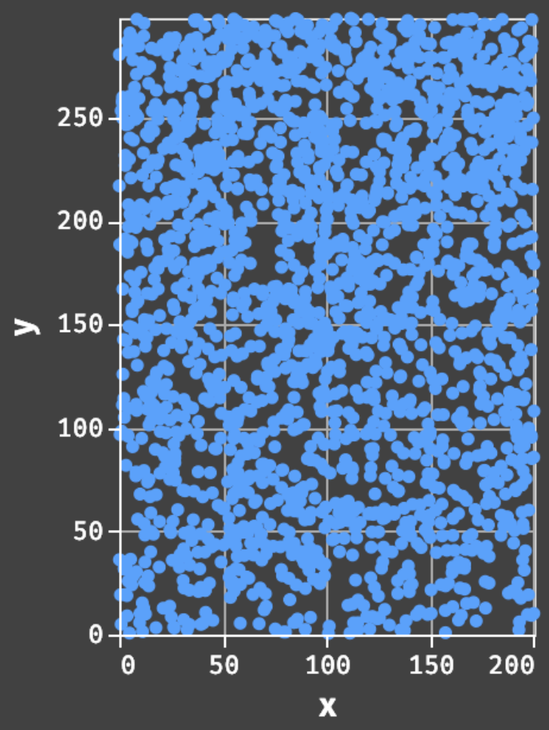

Binning
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0


Modeling
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
-3.0
-2.0
-1.0
0.0
1.0
2.0
3.0

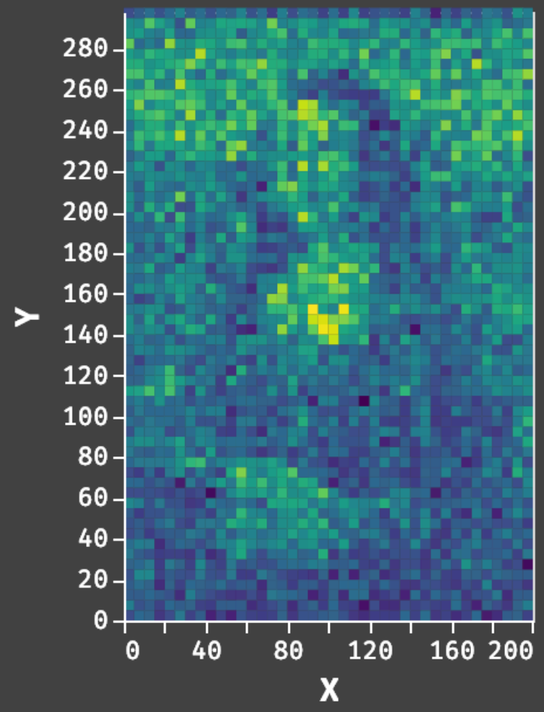
How to **Visualize** a Billion+ Records



Data



Sampling



Binned Aggregation

Decouple the visual complexity from the raw data through aggregation.

Bin > Aggregate (**> Smooth**) **> Plot**

1. Bin Divide data domain into discrete “buckets”

Categories: Already discrete (but watch out for high cardinality)

Numbers: Choose bin intervals (uniform, quantile, ...)

Time: Choose time unit: Hour, Day, Month, etc.

Geo: Bin x, y coordinates *after* cartographic projection

Bin > Aggregate (> Smooth) > Plot

1. Bin Divide data domain into discrete “buckets”

Categories: Already discrete (but watch out for high cardinality)

Numbers: Choose bin intervals (uniform, quantile, ...)

Time: Choose time unit: Hour, Day, Month, etc.

Geo: Bin x, y coordinates *after* cartographic projection

2. Aggregate Count, Sum, Average, Min, Max, ...

Bin > Aggregate (> Smooth) > Plot

1. Bin Divide data domain into discrete “buckets”

Categories: Already discrete (but watch out for high cardinality)

Numbers: Choose bin intervals (uniform, quantile, ...)

Time: Choose time unit: Hour, Day, Month, etc.

Geo: Bin x, y coordinates *after* cartographic projection

2. Aggregate Count, Sum, Average, Min, Max, ...

3. Smooth Optional: smooth aggregates [Wickham ’13]

Bin > Aggregate (> Smooth) > Plot

1. Bin Divide data domain into discrete “buckets”

Categories: Already discrete (but watch out for high cardinality)

Numbers: Choose bin intervals (uniform, quantile, ...)

Time: Choose time unit: Hour, Day, Month, etc.

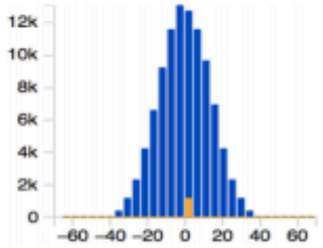
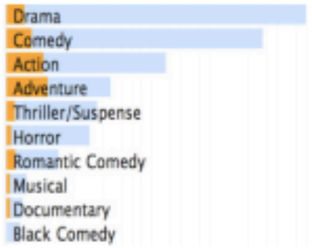
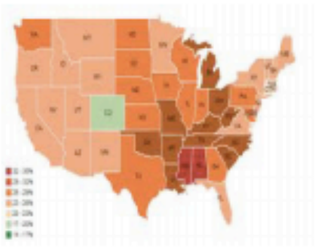
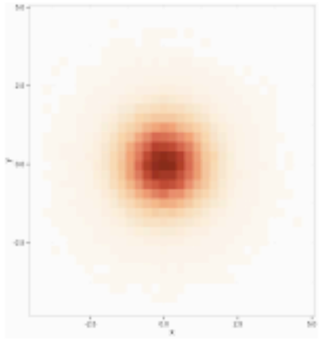
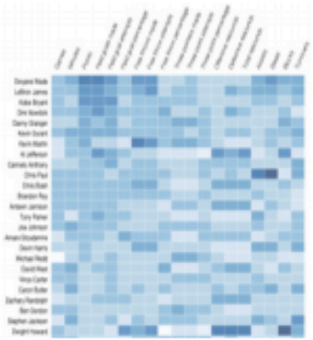
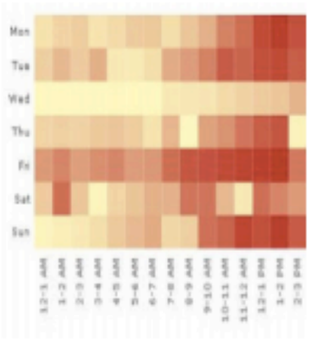
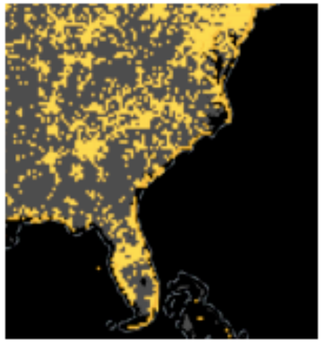
Geo: Bin x, y coordinates *after* cartographic projection

2. Aggregate Count, Sum, Average, Min, Max, ...

3. Smooth Optional: smooth aggregates [Wickham ’13]

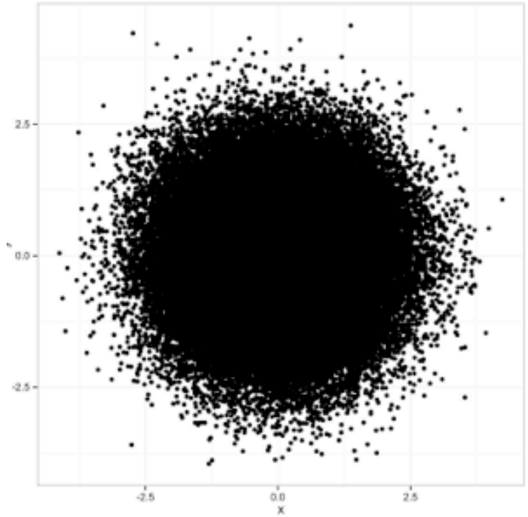
4. Plot Visualize the aggregate values

Binned Plots by Data Type

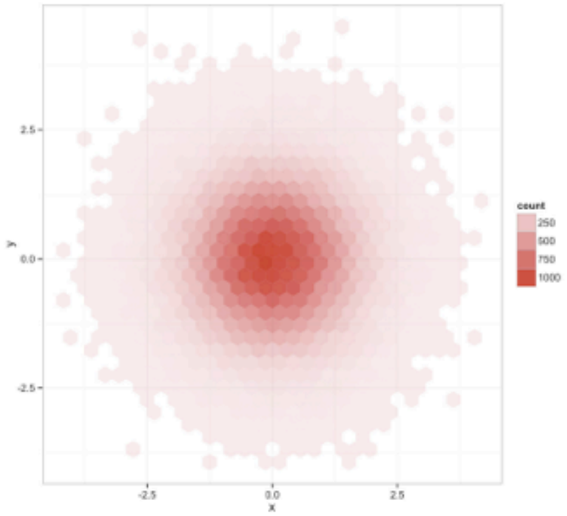
	Numeric	Ordinal	Temporal	Geographic
1D	 Histogram	 Bar Chart	Line Graph / Area Chart	 Choropleth Map
2D	 Binned Scatter Plot	 Heatmap	 Temporal Heatmap	 Geographic Heatmap

Design Subtleties…

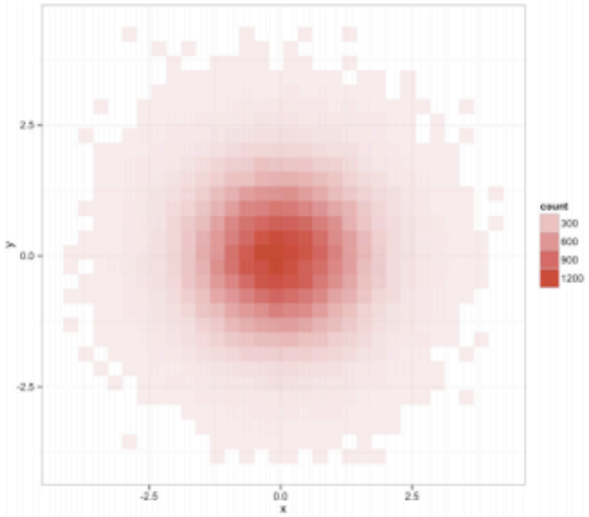
Hexagonal or Rectangular Bins?



100,000 Data Points



Hexagonal Bins

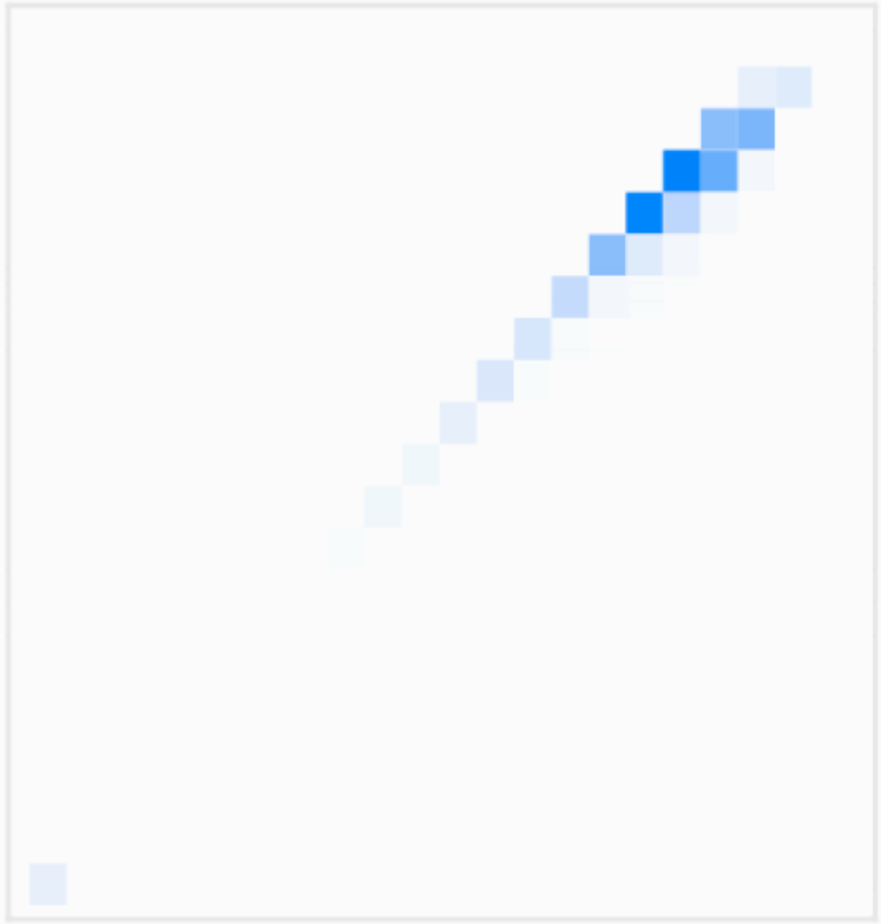


Rectangular Bins

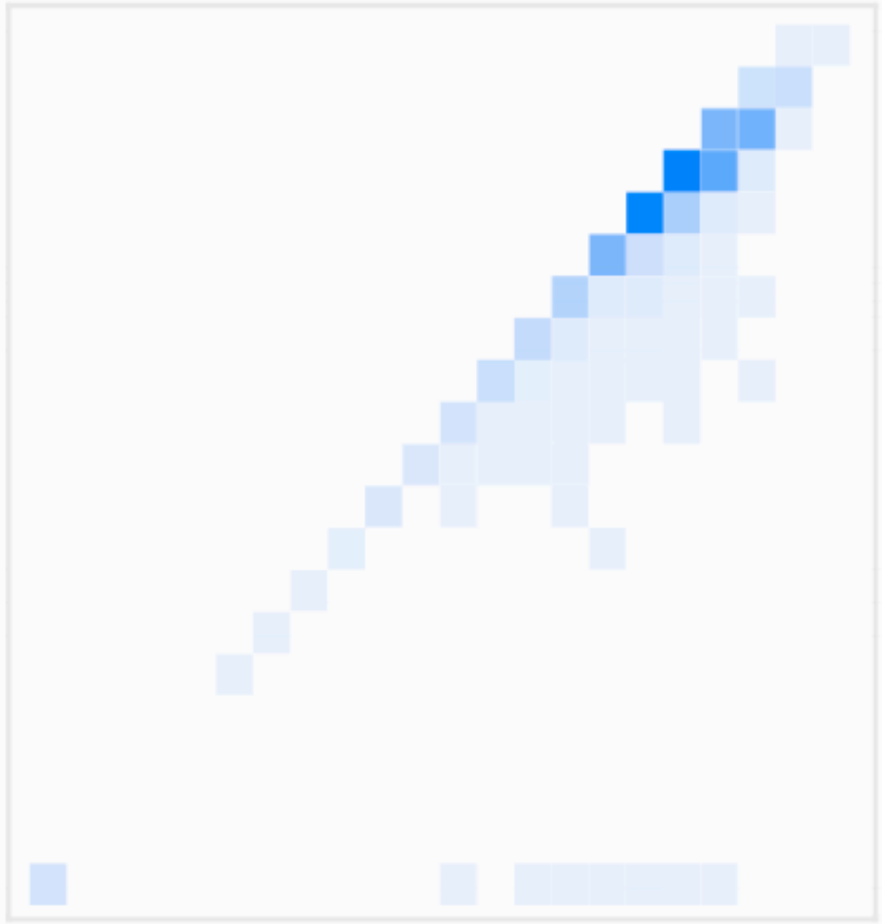
Hex bins better estimate density for 2D plots, but the *improvement is marginal* [Scott 92].

Rectangles support *reuse* and *visual queries*.

Color Scale: Discontinuity after Zero



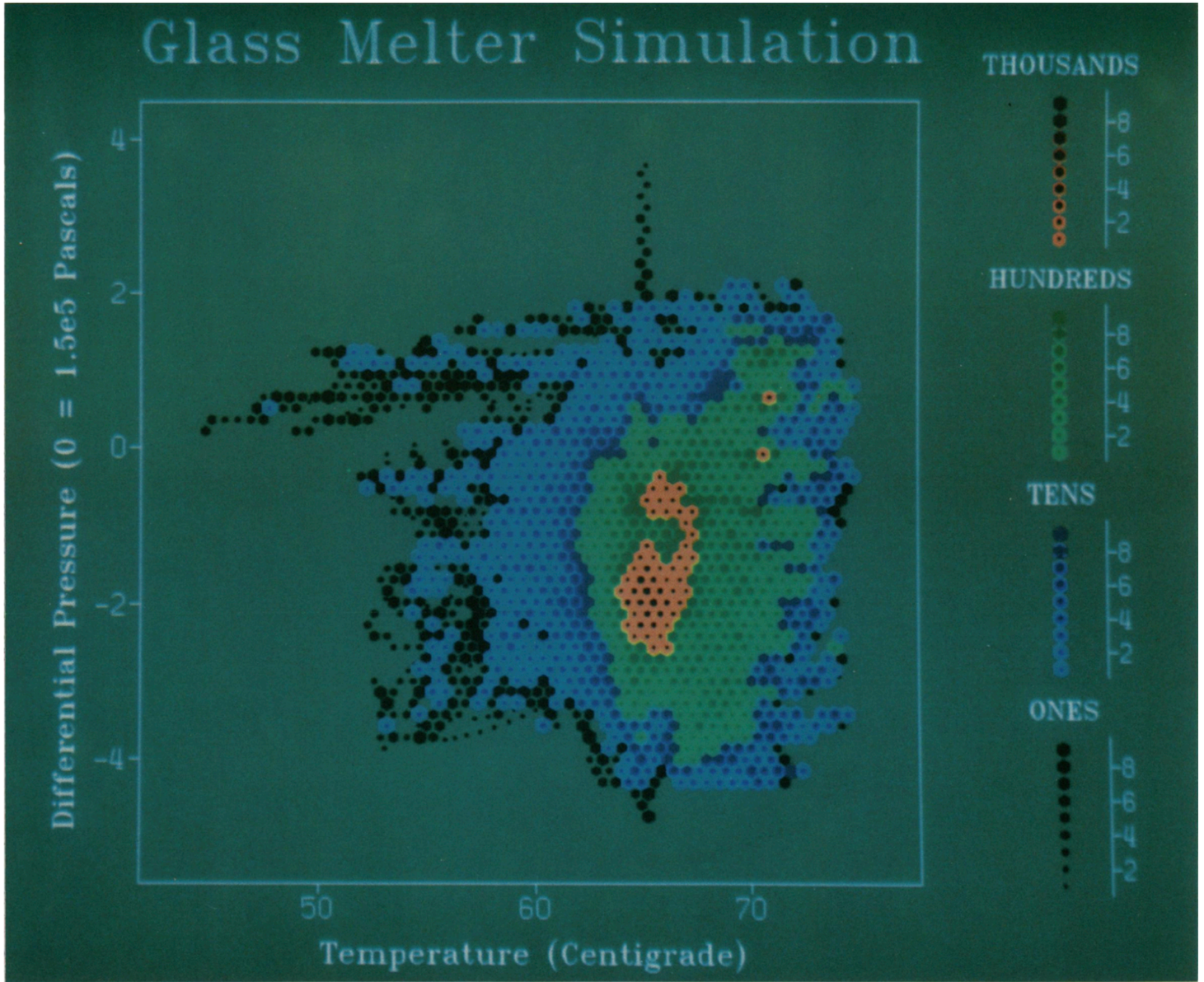
Standard Color Ramp
Counts near zero are white.



Add Discontinuity after Zero
Counts near zero remain visible.

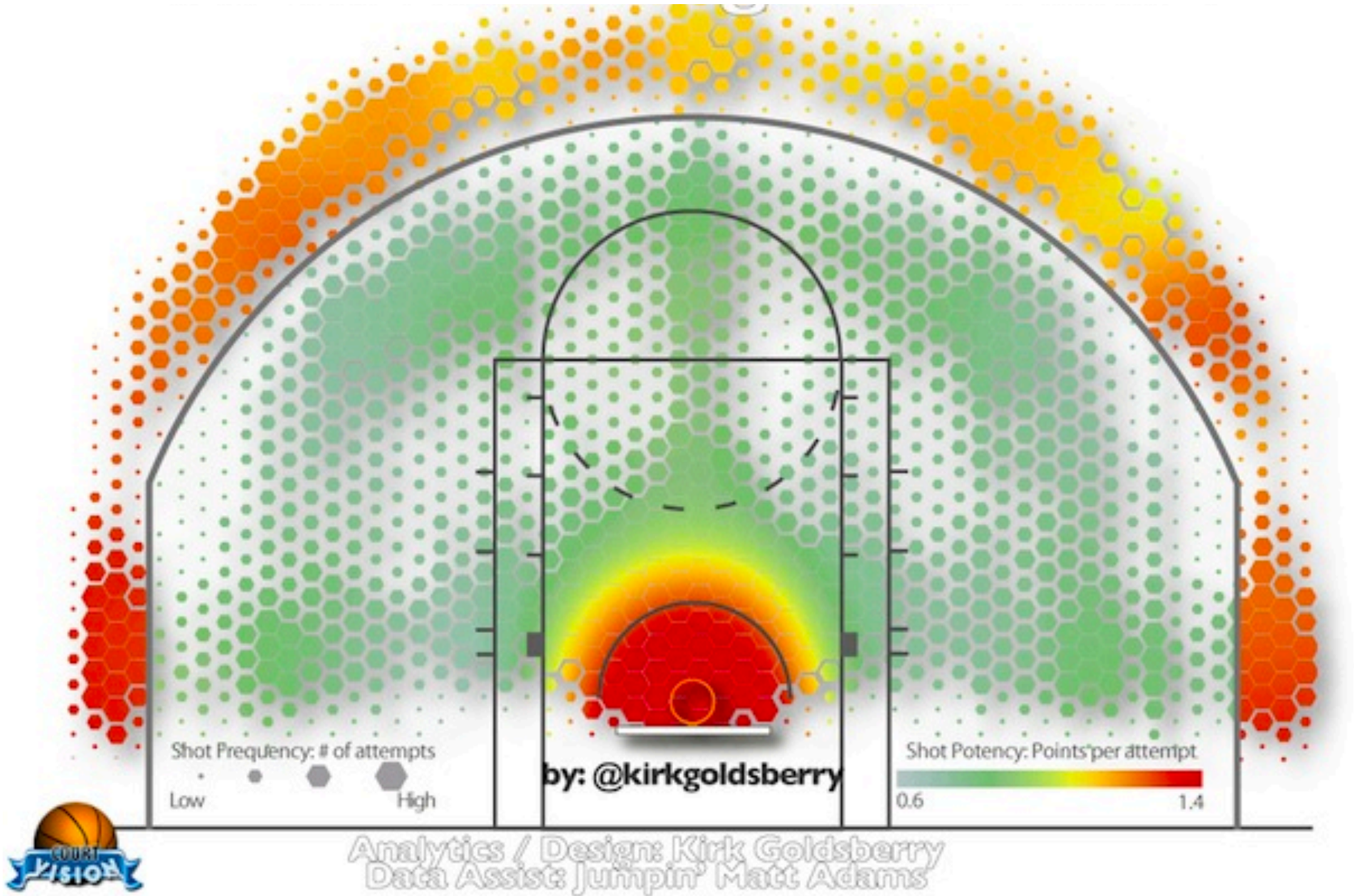
Examples

Example: Binned Scatter Plots



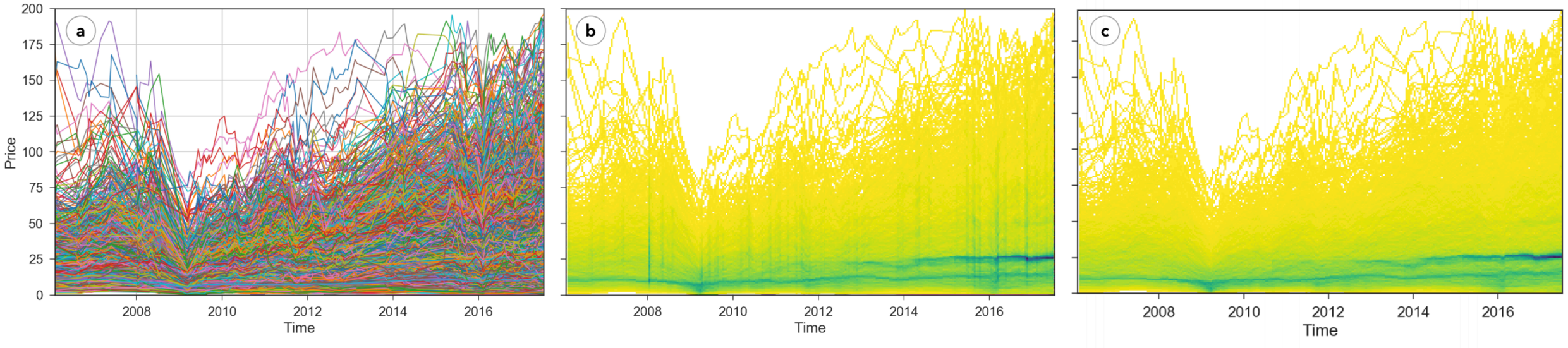
Scatterplot Matrix
Techniques for Large N
[Carr et al. '87]

Example: Basketball Shot Chart



NBA Shooting 2011-12
[Goldsberry]

Example: Density Line Chart [Moritz & Fisher]

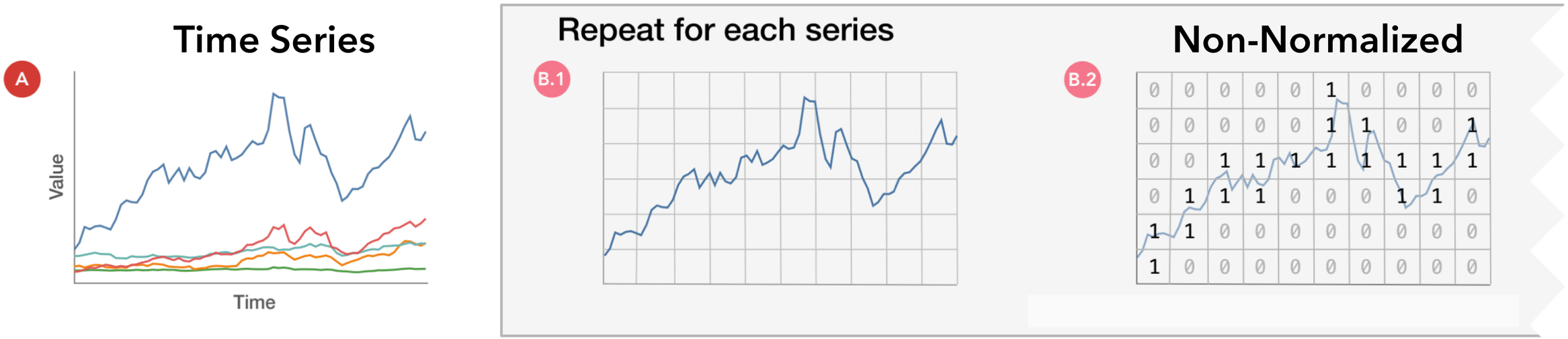


Line Chart | Non-Normalized Heatmap | Normalized "DenseLines"

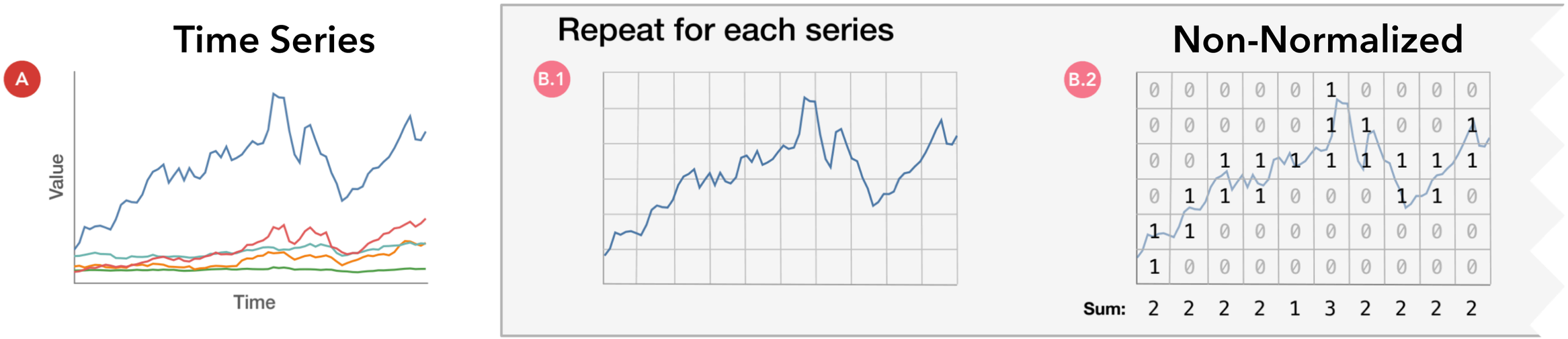
The non-normalized heatmap suffers from artifacts, seen as vertical stripes.

Binned charts convey high points across the top, a collective dip in stocks during the crash of 2008, and two distinct bands of $25 and $15 stocks.

Example: Density Line Chart [Moritz & Fisher]



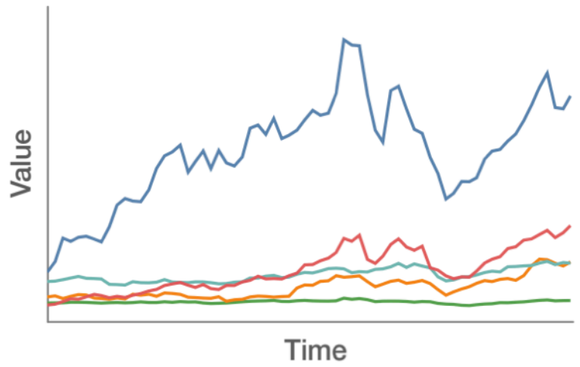
Example: Density Line Chart [Moritz & Fisher]



Example: Density Line Chart [Moritz & Fisher]

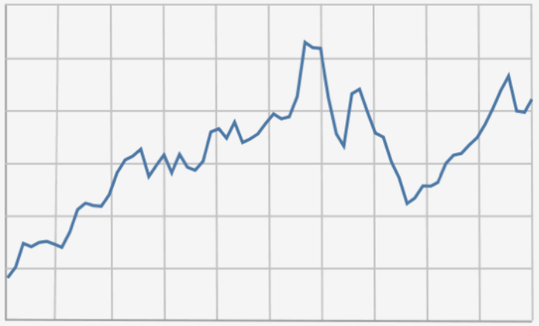
Time Series

A



Repeat for each series

B.1



Non-Normalized

B.2

0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1
0	1	1	1	0	0	0	1	1	0
1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
Sum:	2	2	2	2	1	3	2	2	2	2

B.3

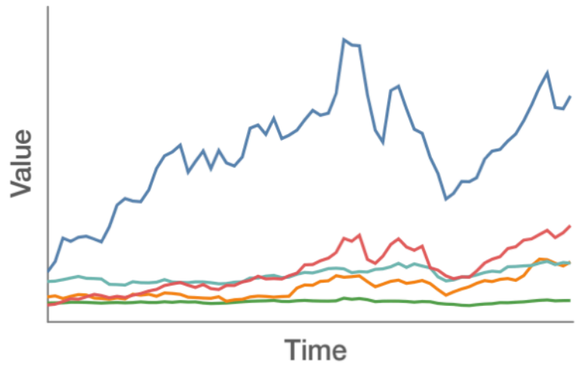
0	0	0	0	0	0.3	0	0	0	0
0	0	0	0	0	0.3	0.5	0	0	0.5
0	0	0.5	0.5	0.5	0.3	0.5	0.5	0.5	0.5
0	0.5	0.5	0.5	0	0	0	0.5	0.5	0
0.5	0.5	0	0	0	0	0	0	0	0
0.5	0	0	0	0	0	0	0	0	0

Approx. Arc-Length Normalized

Example: Density Line Chart [Moritz & Fisher]

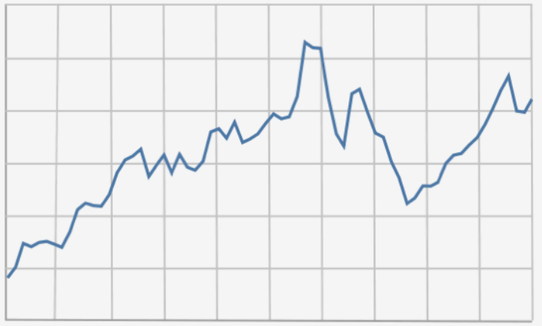
Time Series

A



Repeat for each series

B.1



Non-Normalized

B.2

0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1
0	1	1	1	0	0	0	1	1	0
1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
Sum:	2	2	2	2	1	3	2	2	2	2

Approx. Arc-Length Normalized

B.3

0	0	0	0	0	0.3	0	0	0	0
0	0	0	0	0	0.3	0.5	0	0	0.5
0	0	0.5	0.5	0.5	0.3	0.5	0.5	0.5	0.5
0	0.5	0.5	0.5	0	0	0	0.5	0.5	0
0.5	0.5	0	0	0	0	0	0	0	0
0.5	0	0	0	0	0	0	0	0	0

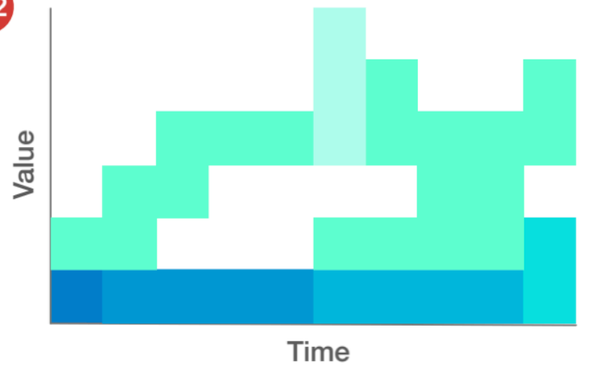
Aggregate

C.1

					0.3				
					0.3	0.5			0.5
		0.5	0.5	0.5	0.3	0.5	0.5	0.5	0.5
	0.5	0.5					0.5	0.5	
0.5	0.5					0.5	0.5	0.5	0.5	2
4.5	4	4	4	4	3.5	3.5	3.5	3.5	2

Color

C.2



2. Enabling Real-Time Interaction

Interactive Scalability Strategies

1. Query Database
2. Client-Side Indexing / Data Cubes
3. Prefetching
4. Approximation

Interactive Scalability Strategies

1. Query Database Offload to a scalable backend

Tableau, for example, issues aggregation queries.

Analytical databases are designed for fast, parallel execution.

But round-trip queries to the DB may still be too slow…

2. Client-Side Indexing / Data Cubes

3. Prefetching

4. Approximation

Interactive Scalability Strategies

1. **Query Database**
2. **Client-Side Indexing / Data Cubes** Query data summaries

Build sorted indices or data cubes to quickly re-calculate aggregations as needed on the client.

3. **Prefetching**
4. **Approximation**

Interactive Scalability Strategies

1. **Query Database**
2. **Client-Side Indexing / Data Cubes**
3. **Prefetching** Request data *before* it is needed

Reduce latency by speculatively querying for data before it is needed. Requires prediction models to guess what is needed.

4. **Approximation**

Interactive Scalability Strategies

1. **Query Database**
2. **Client-Side Indexing / Data Cubes**
3. **Prefetching**
4. **Approximation** Give fast, approximate answers

Reduce latency by computing aggregates on a sample, ideally with approximation bounds characterizing the error.

Interactive Scalability Strategies

1. **Query Database**
2. **Client-Side Indexing / Data Cubes**
3. **Prefetching**
4. **Approximation**

These strategies are **not** mutually exclusive!

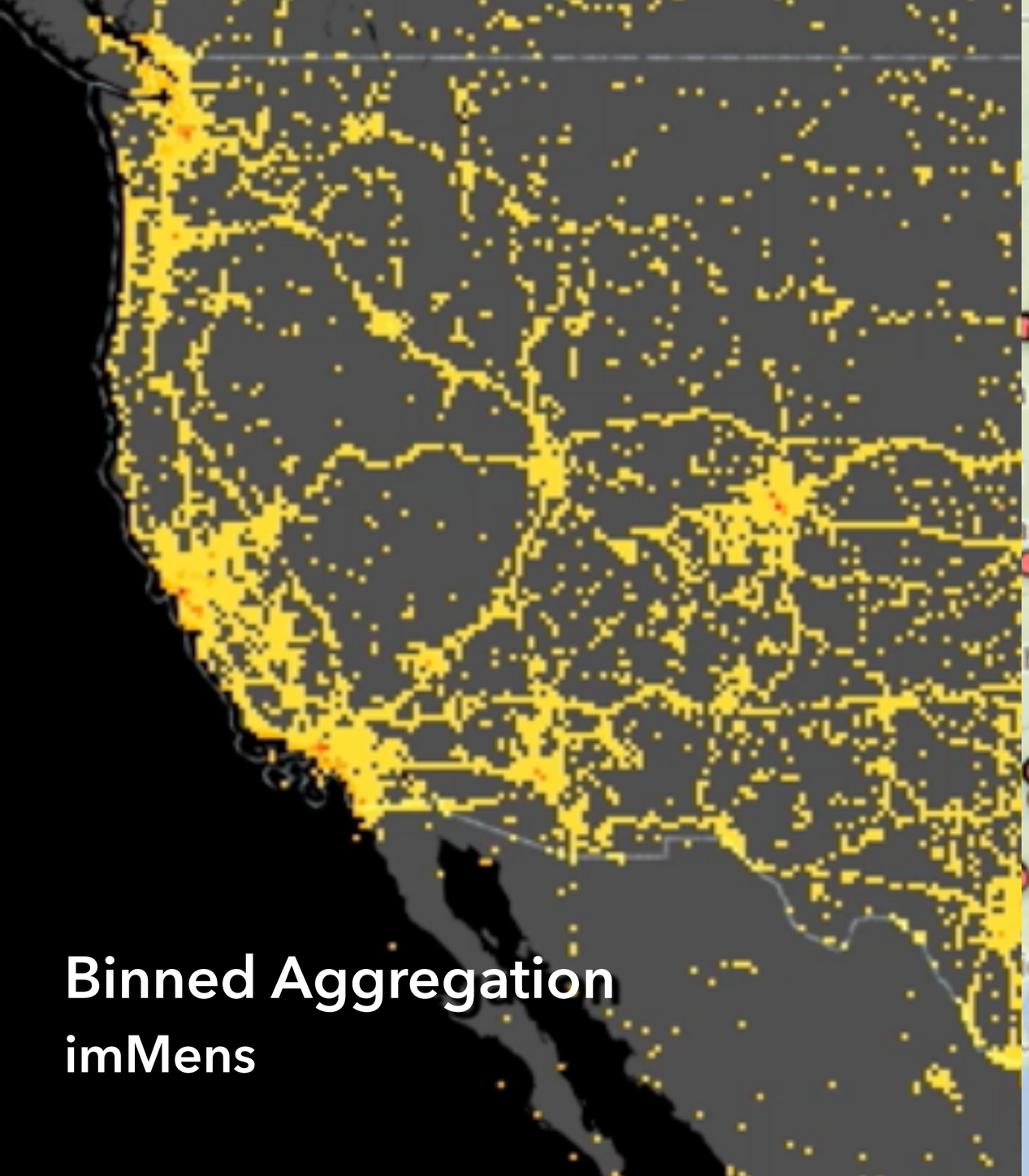
Systems can apply them in tandem.

imMens

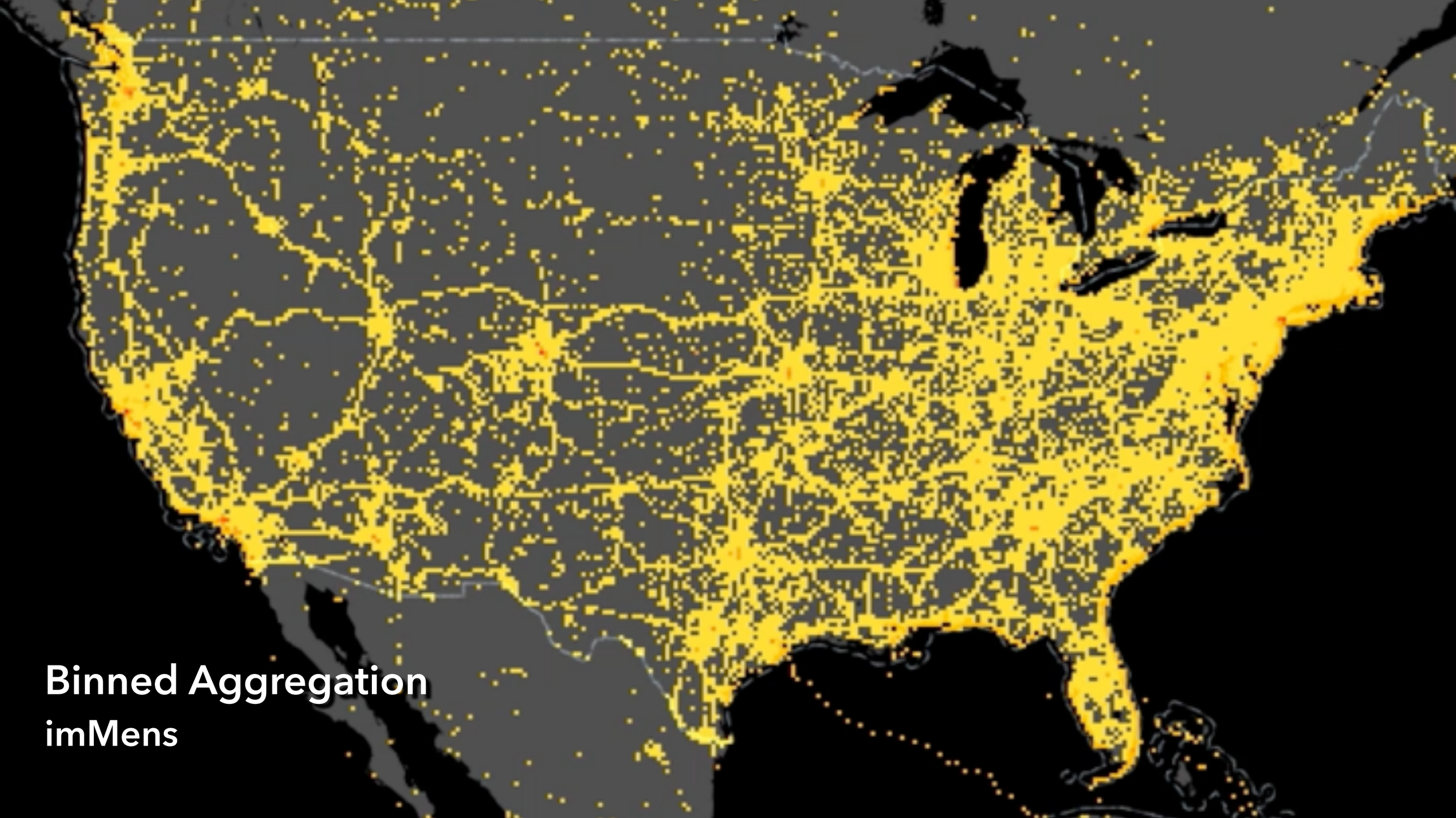
[Liu, Jiang & Heer ’13]

Strategies: Client-Side Data Cubes

Montana
North Dakota
Minnesota
South Dakota
Wisconsin
Michigan
Toronto
Montreal
Maine
Oregon
Idaho
New York
Illinois
Ohio
Pennsylvania
Nevada
Utah
Missouri
West Virginia
Kentucky
Virginia
Tennessee
Oklahoma
Delaware
Maryland
New Mexico
Mississippi
Alabama
Georgia
Dallas
Texas
Houston
San Antonio
Phoenix
San Diego
Gulf of California
Monterrey
Mexico
Havana
Sampling
Google Fusion Tables

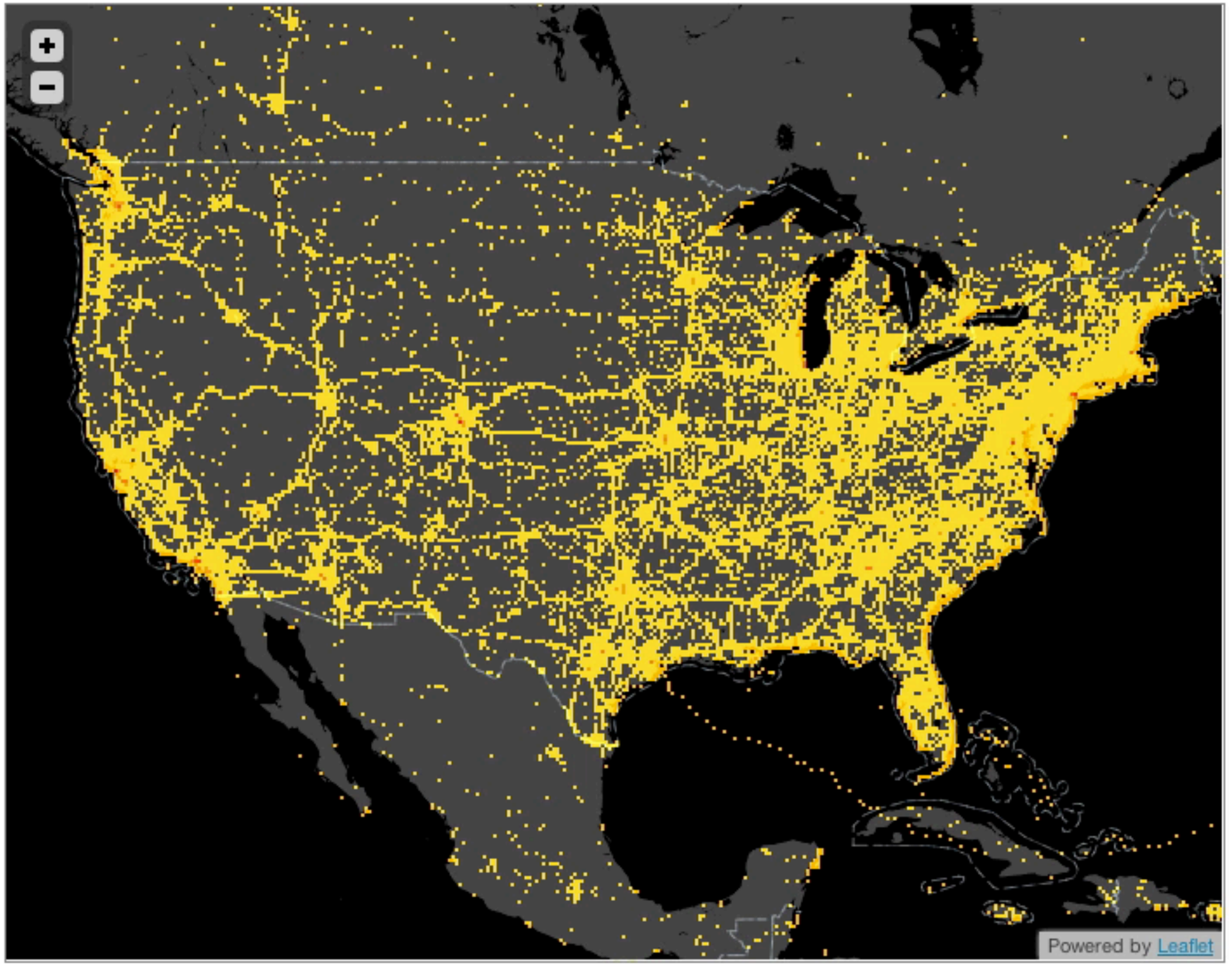

Binned Aggregation
imMens

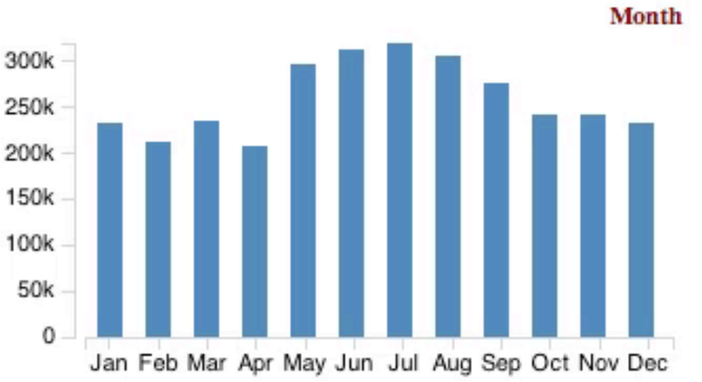
Minnesota
Wisconsin
Michigan
Illinois
Indiana
Ohio
Missouri
Kentucky
Tennessee
Mississippi
Alabama
Georgia
Louisiana
Dallas
Houston
Mexico
Havana
Toronto
Montreal
Maine
New York
Pennsylvania
West Virginia
Virginia
Carolina
Delaware
Maryland
District of Columbia
Connecticut
Sampling
Google Fusion Tables

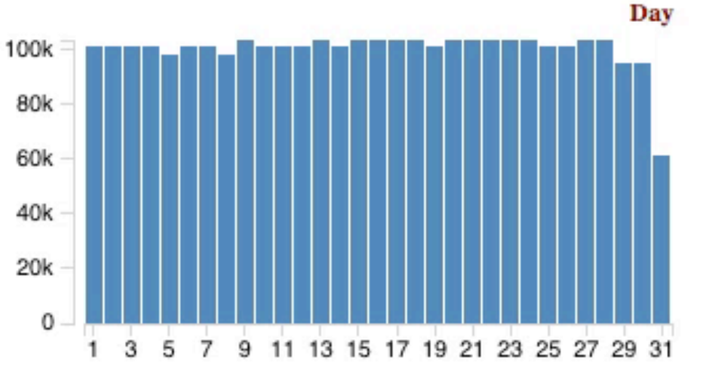


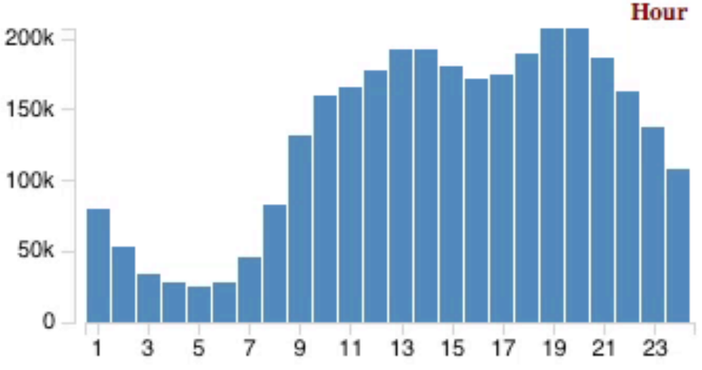
Binned Aggregation

imMens


Powered by Leaflet


Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

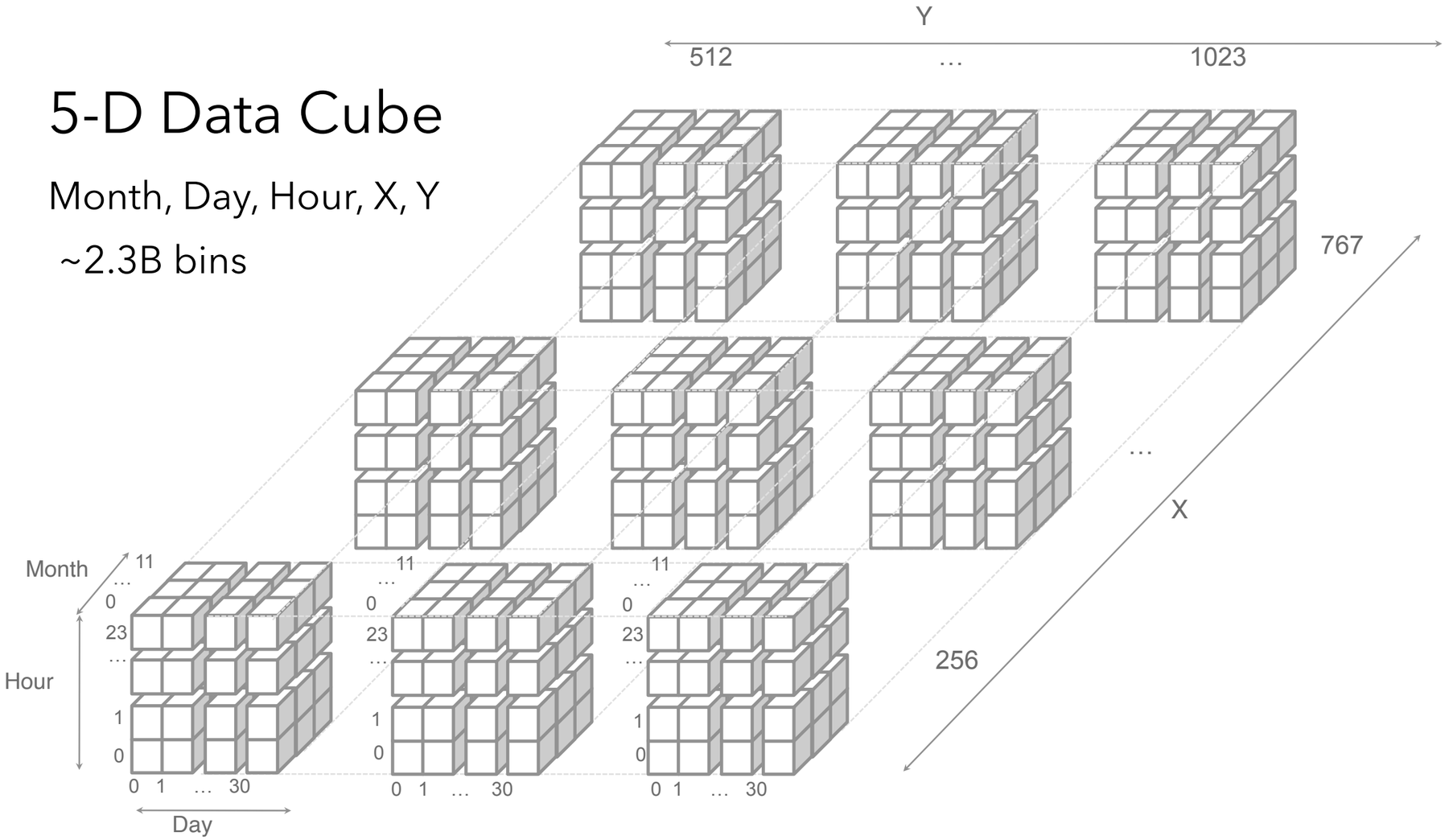

Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31


Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23

5-D Data Cube

Month, Day, Hour, X, Y

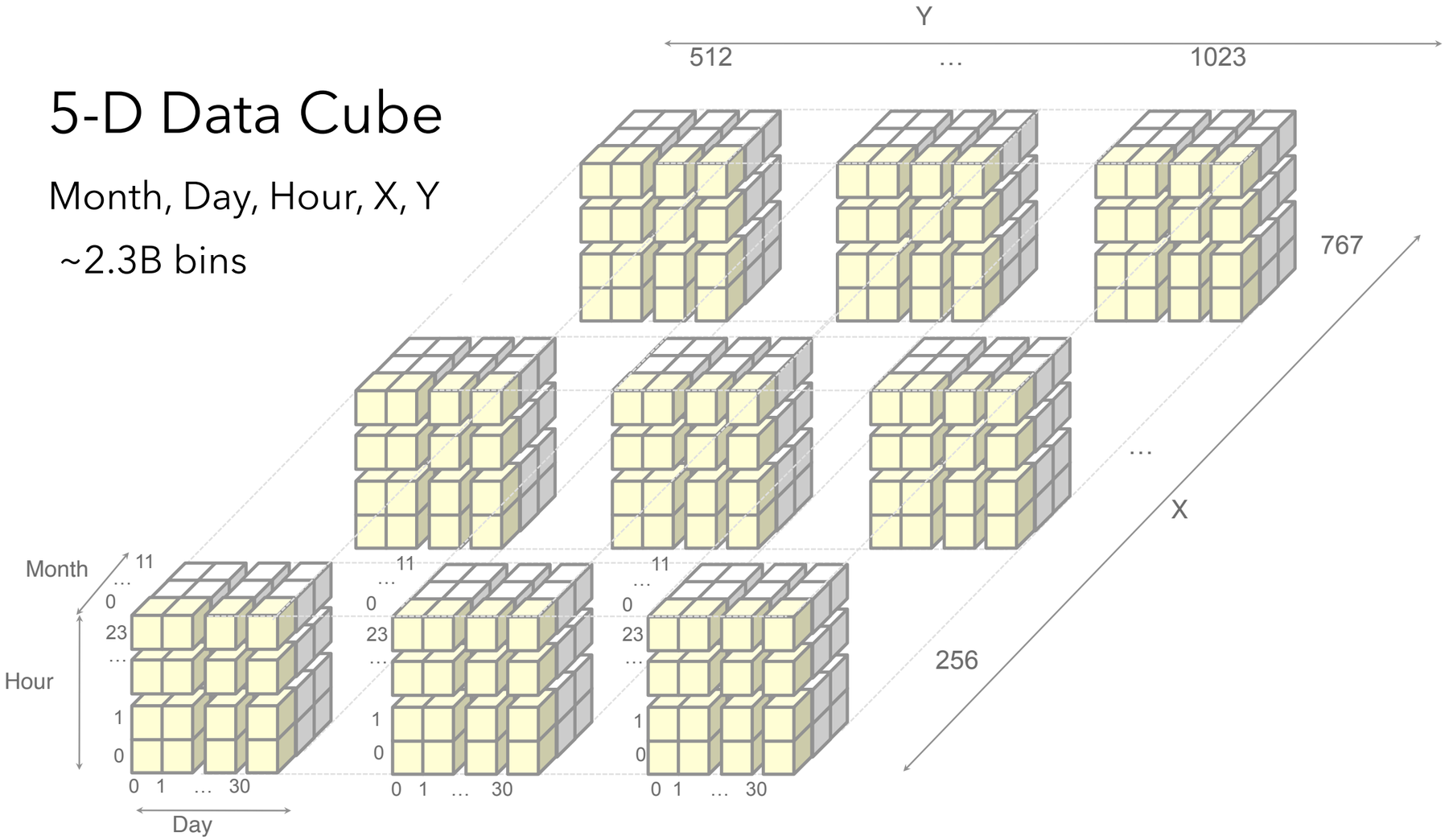
~2.3B bins

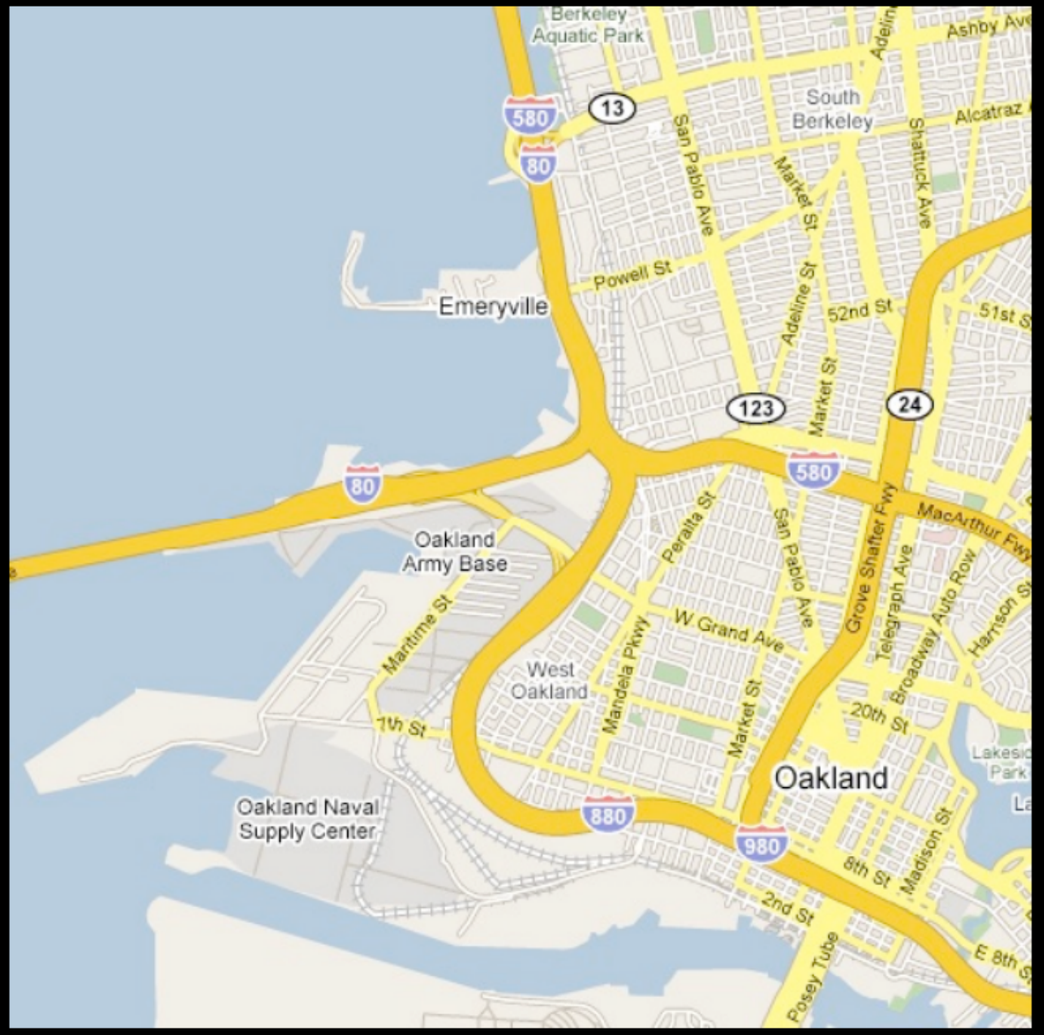


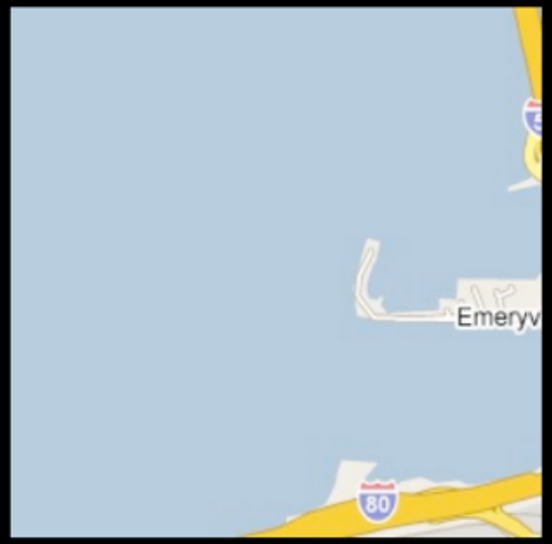
5-D Data Cube

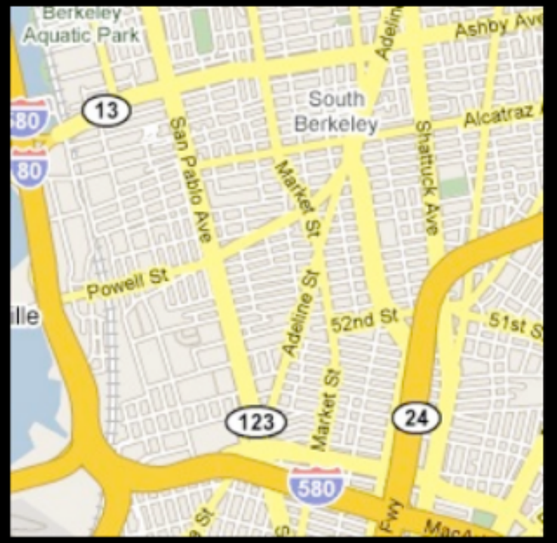
Month, Day, Hour, X, Y

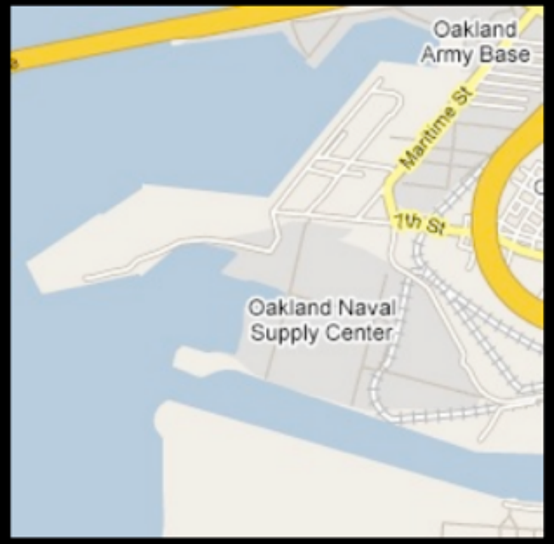
~2.3B bins

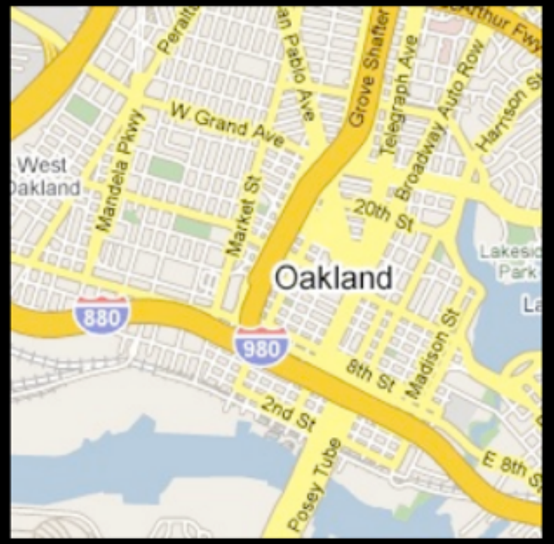



Berkeley Aquatic Park
Ashby Ave
South Berkeley
Alcatraz
580
13
80
San Pablo Ave
Market St
Shattuck Ave
Powell St
Emeryville
Adeline St
52nd St
51st St
123
24
Market St
580
80
Oakland Army Base
Peralta St
San Pablo Ave
Grove Shafter Fwy
MacArthur Fwy
Telegraph Ave
Broadway Auto Row
Harrison St
Maritime St
W Grand Ave
West Oakland
Mandela Pkwy
Market St
20th St
7th St
Oakland
Oakland Naval Supply Center
880
980
8th St
Madison St
2nd St
Posey Tube
E 8th St


Emeryv
80

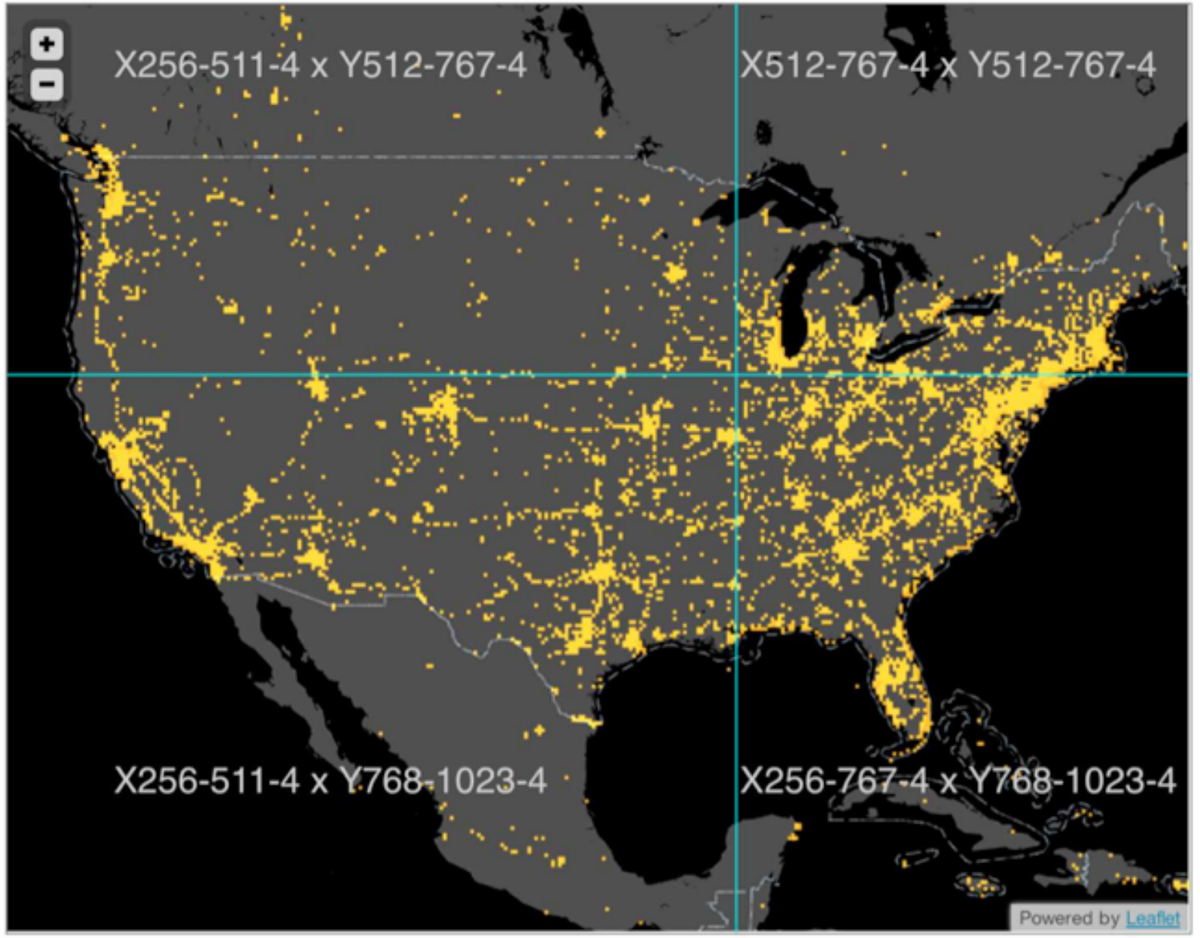

Berkeley
Aquatic Park
Ashby Ave
13
80
South
Berkeley
Alcatraz
San Pablo Ave
Market St
Shattuck Ave
Powell St
Adeline St
52nd St
51st St
123
Market St
24
580


Oakland
Army Base
Maritime St
7th St
Oakland Naval
Supply Center

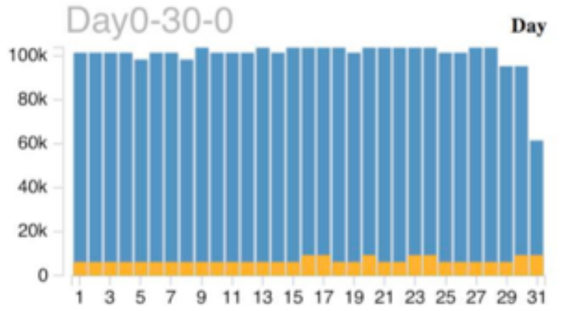

Peralta
San Pablo Ave
Grove Shafter
MacArthur Fwy
W Grand Ave
Telegraph Ave
Broadway Auto Row
Harrison St
West
Oakland
Mandela Pkwy
Market St
20th St
Oakland
880
980
8th St
Madison St
2nd St
E 8th St
Posey Tube

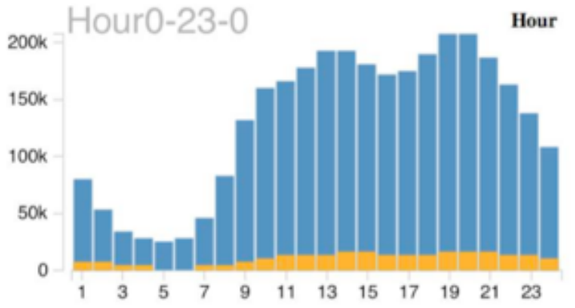
Multivariate Data Tiles

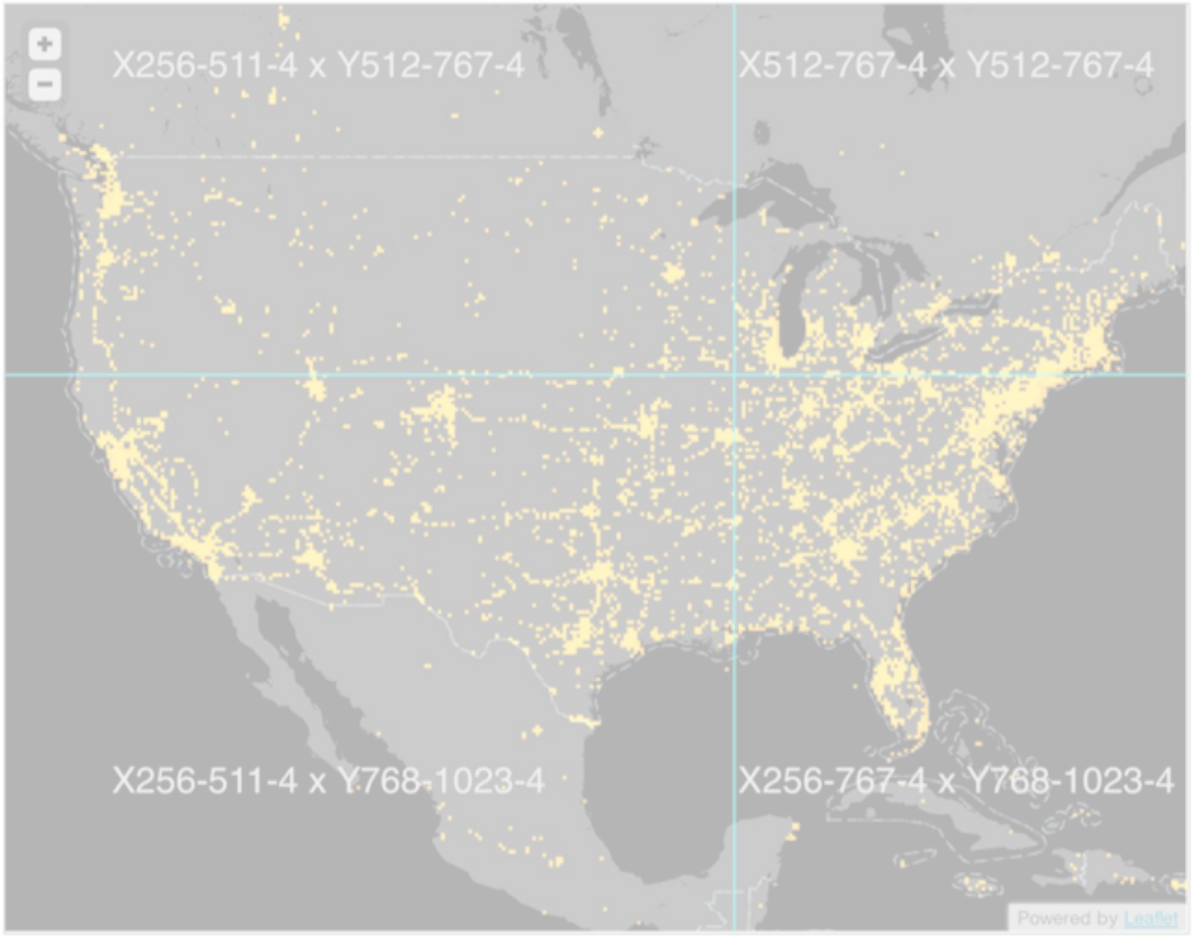
1. Send data, not pixels
2. Embed multi-dim data


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

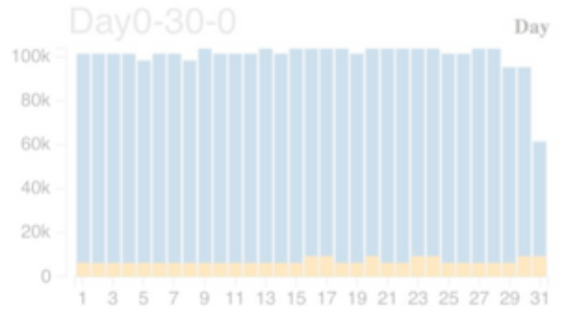
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

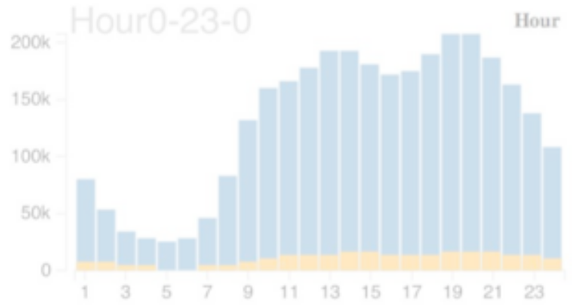

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

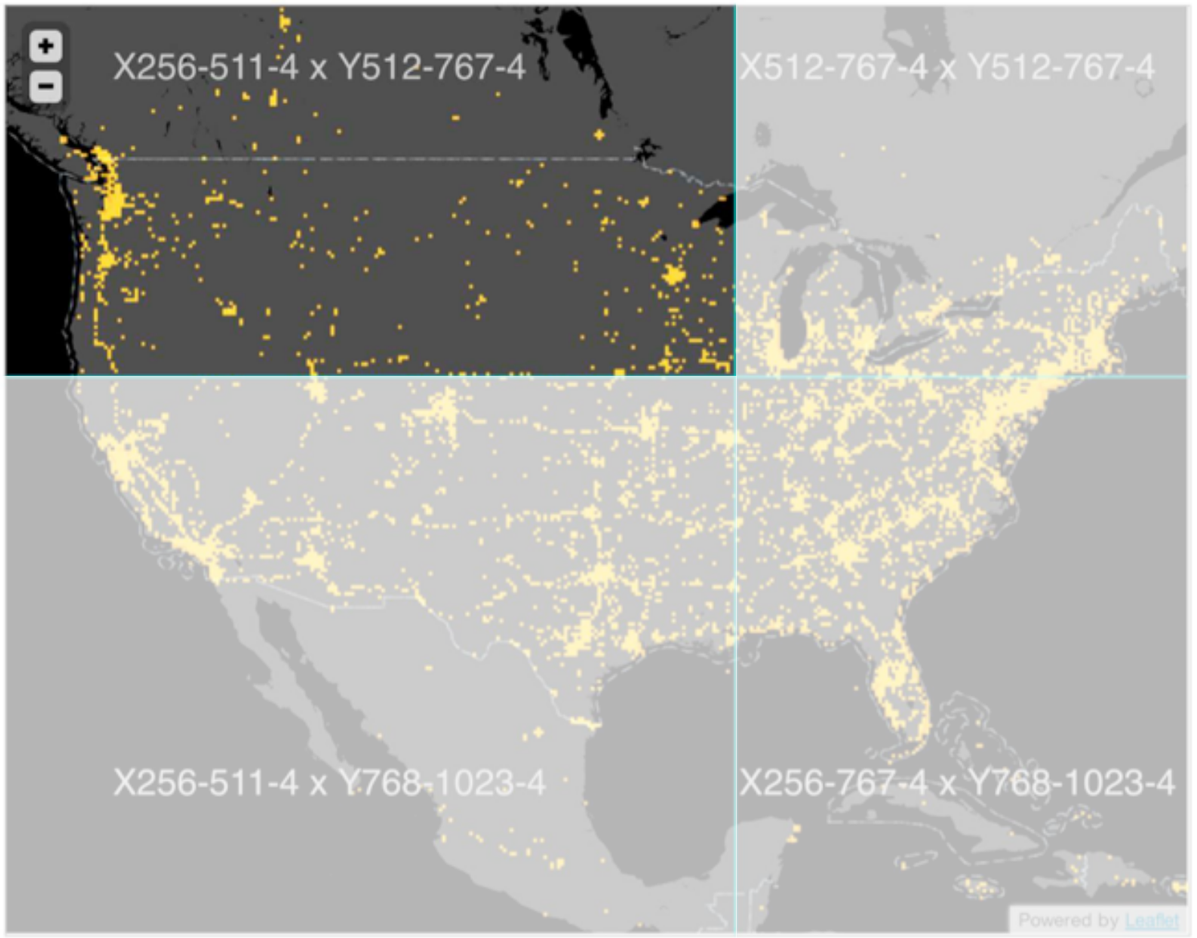

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

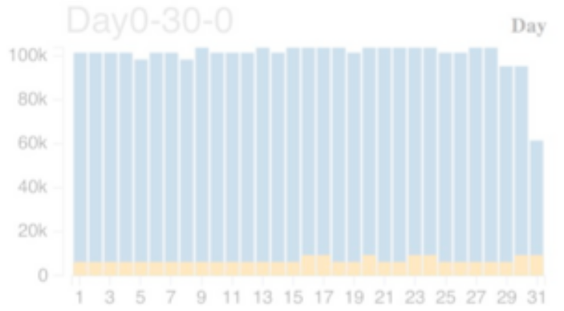
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

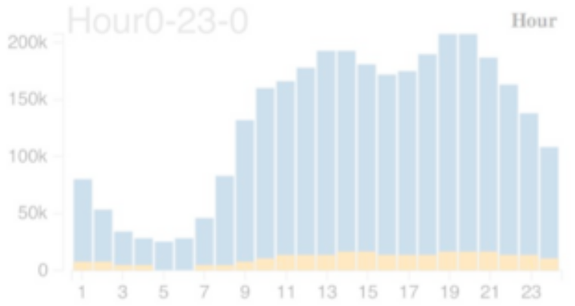

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

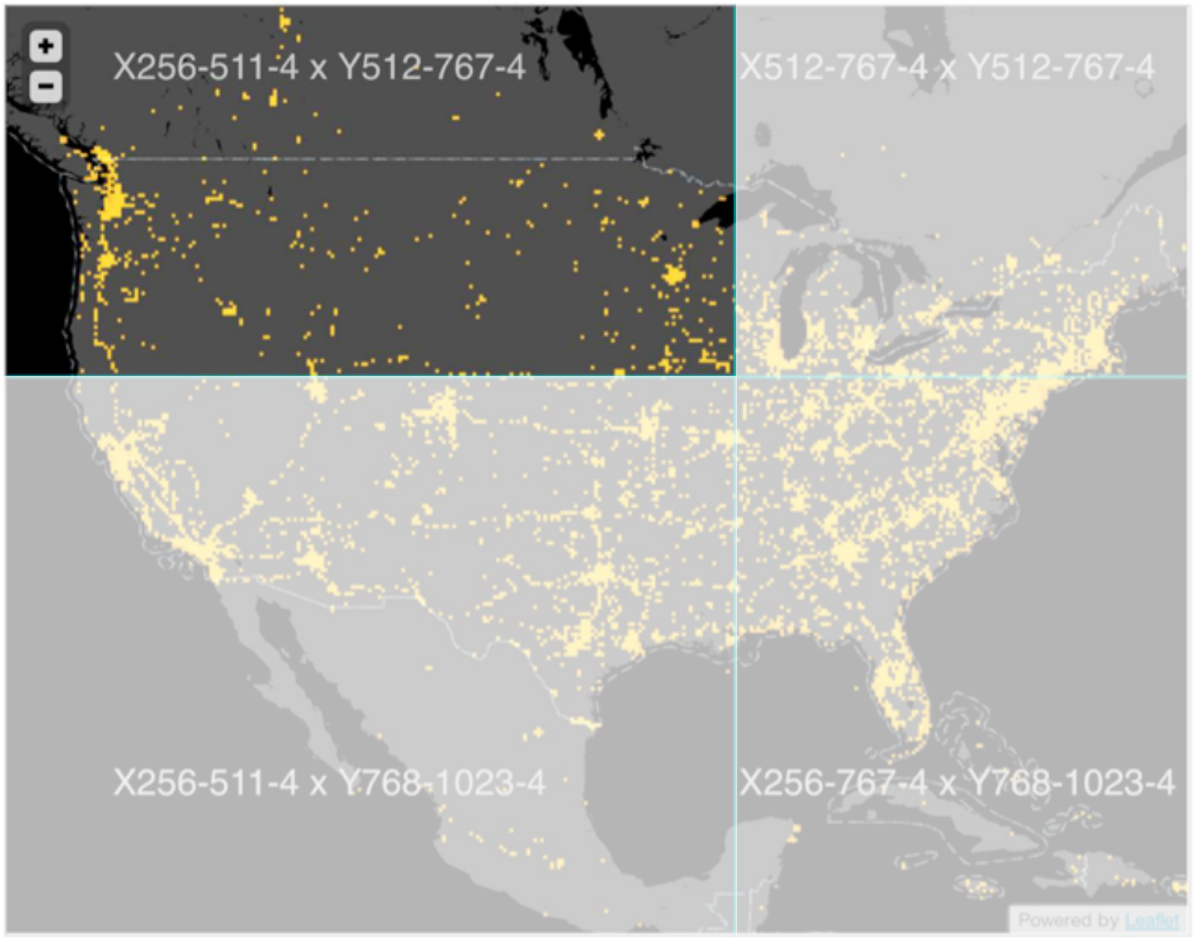

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

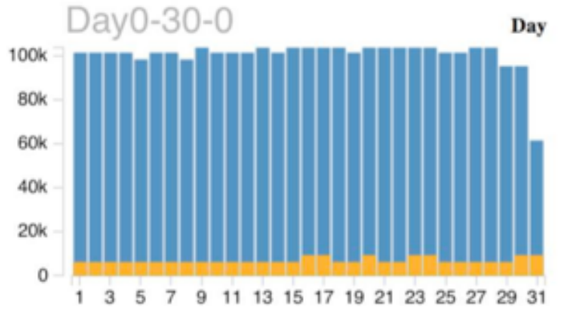
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

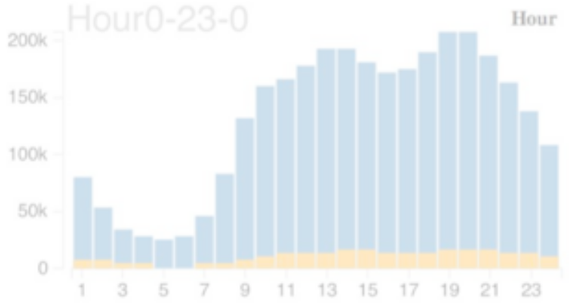

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

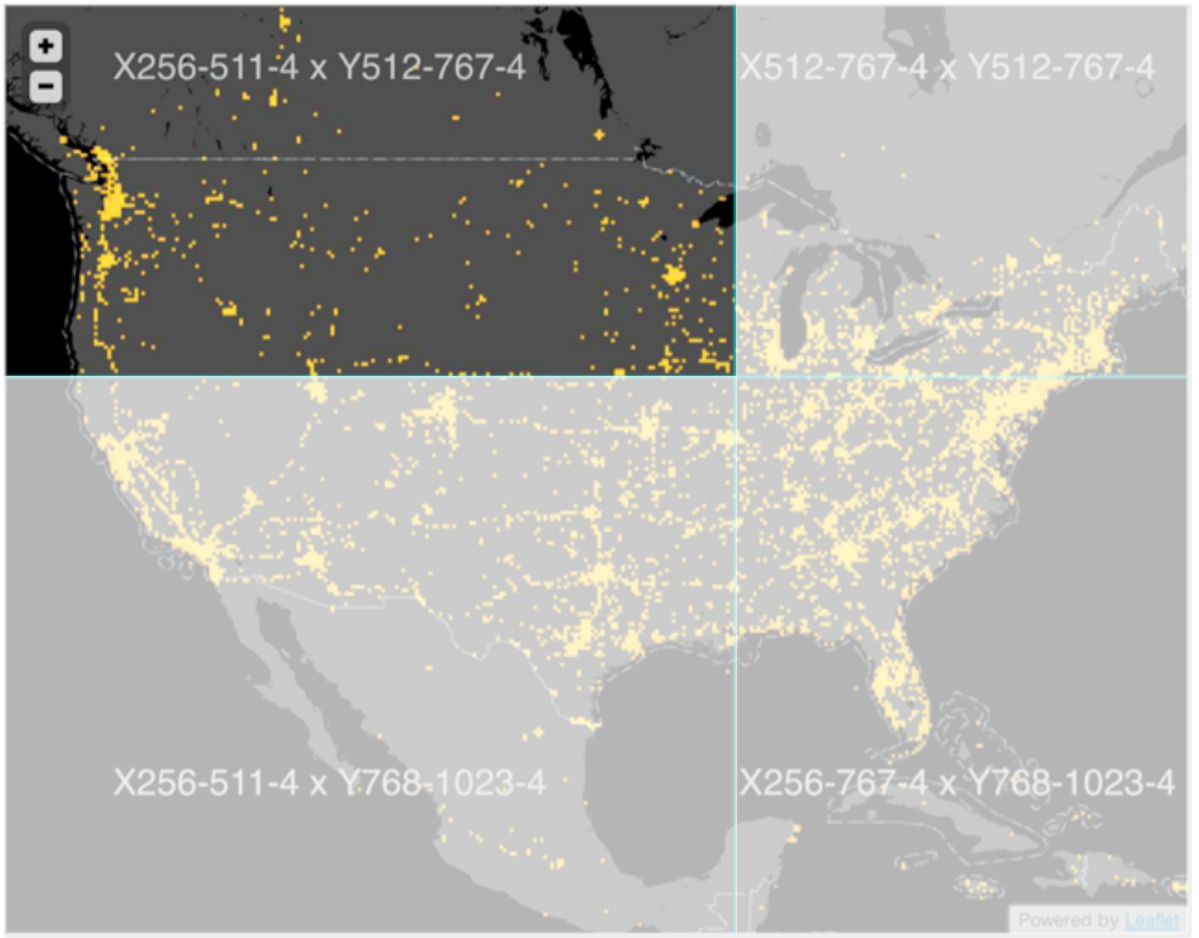

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

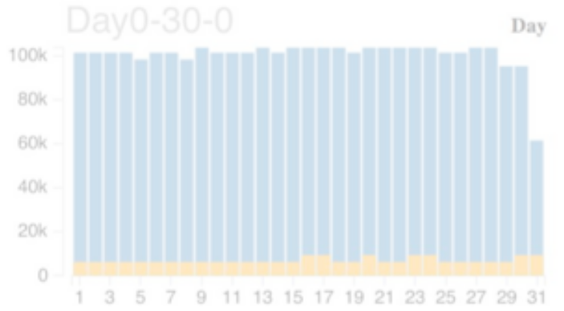
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

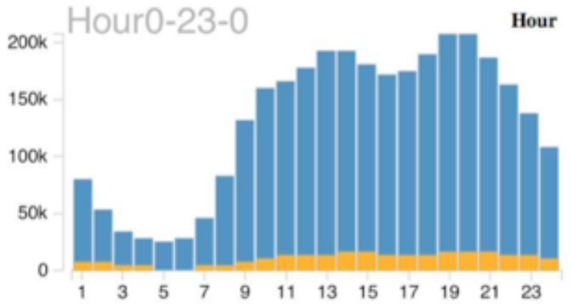

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

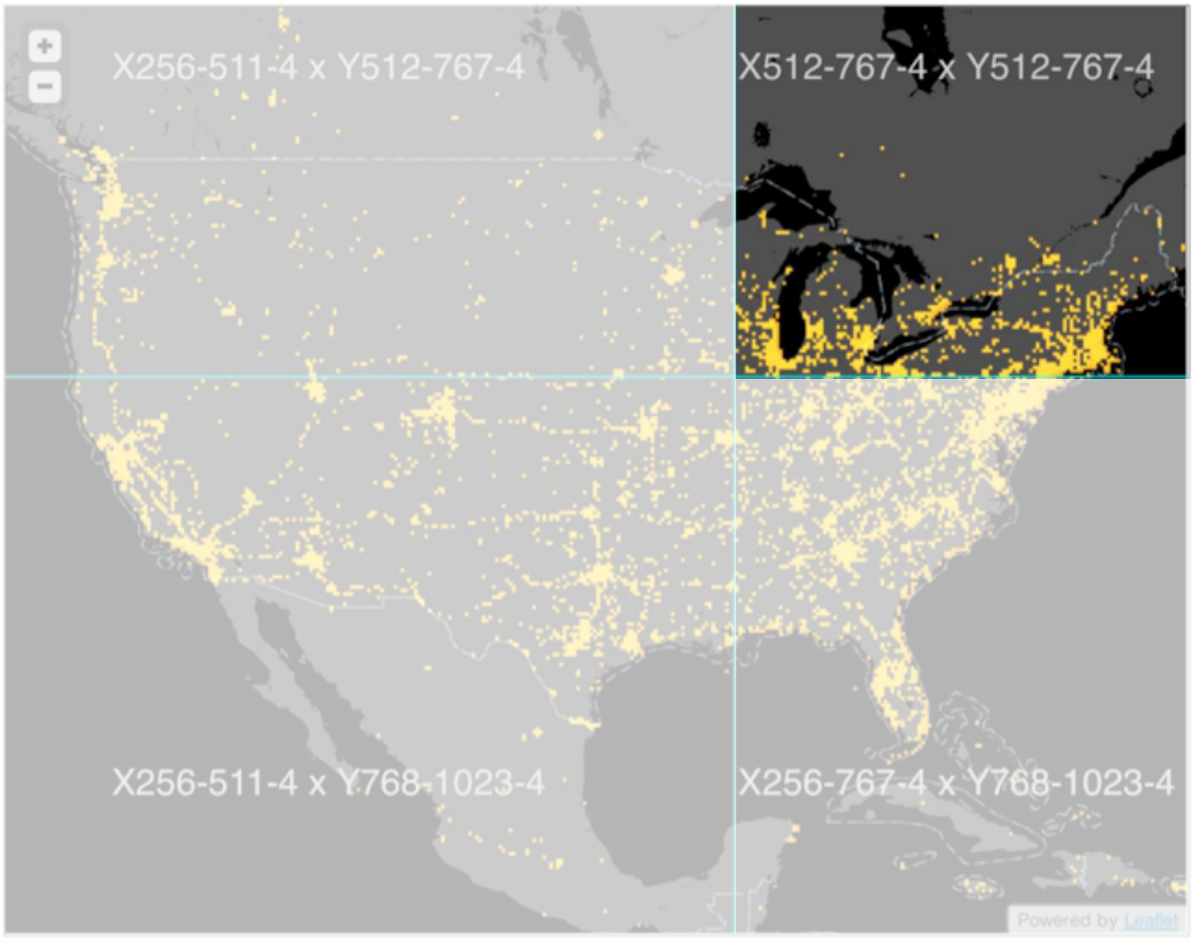

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

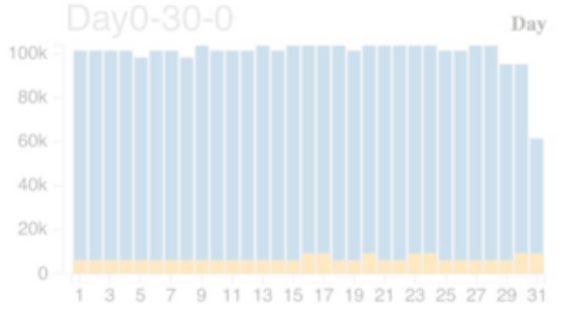
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

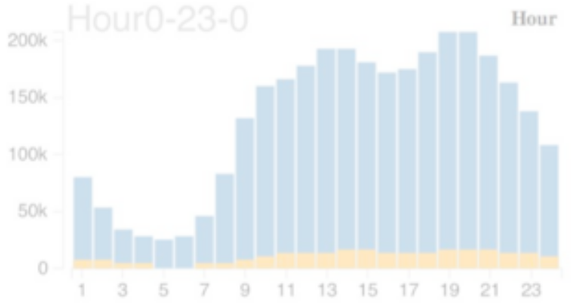

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

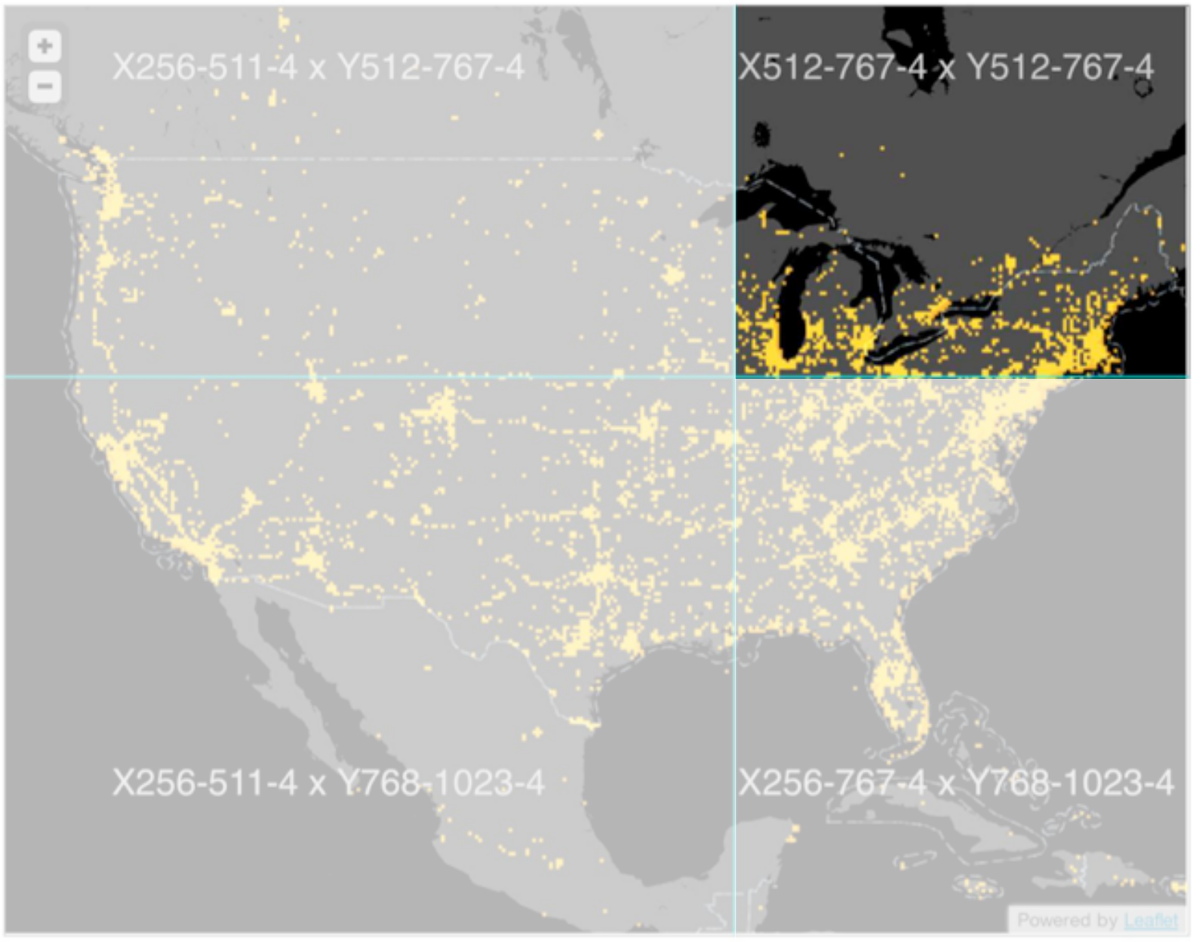

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

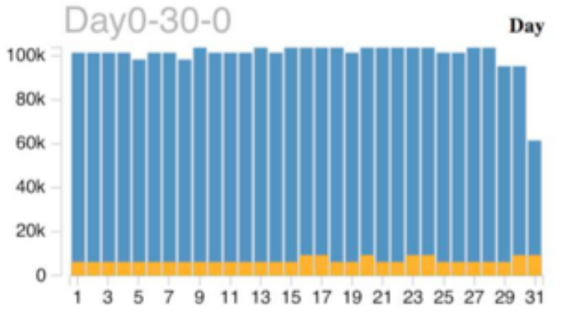
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

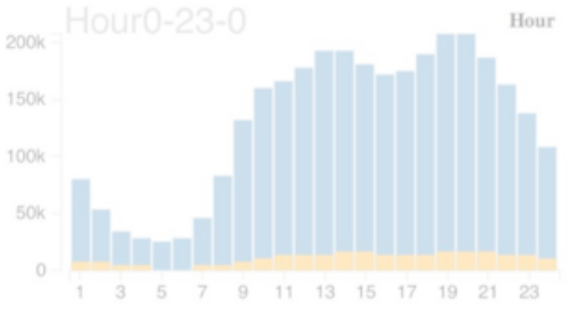

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

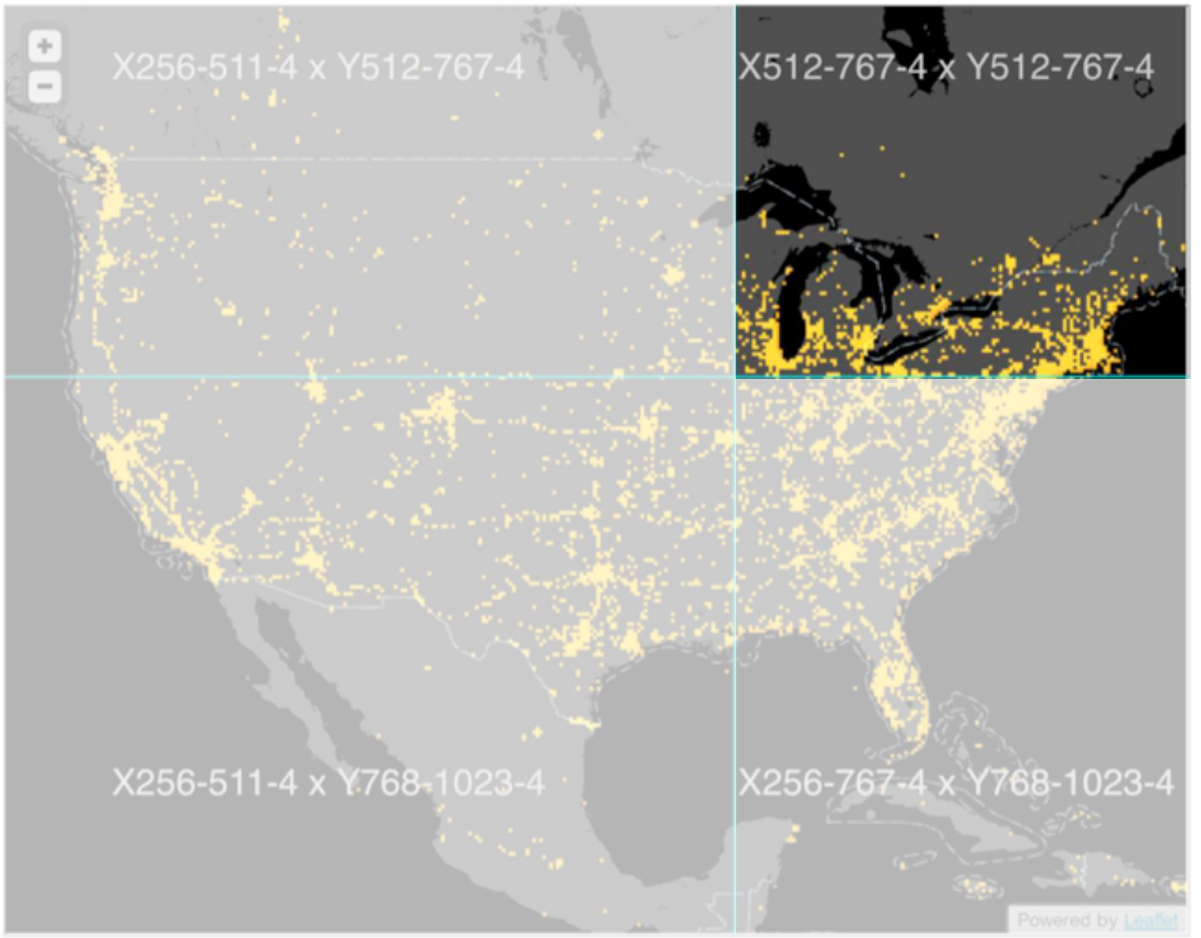

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

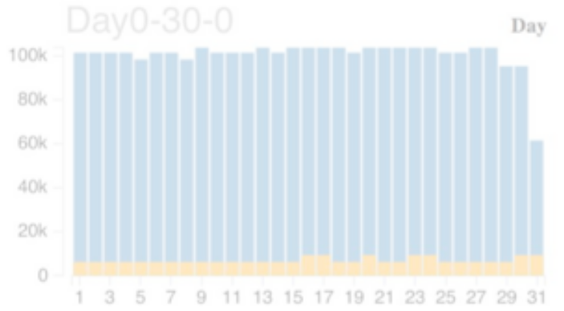
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

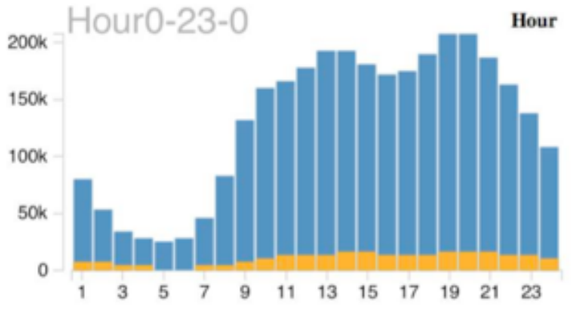

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

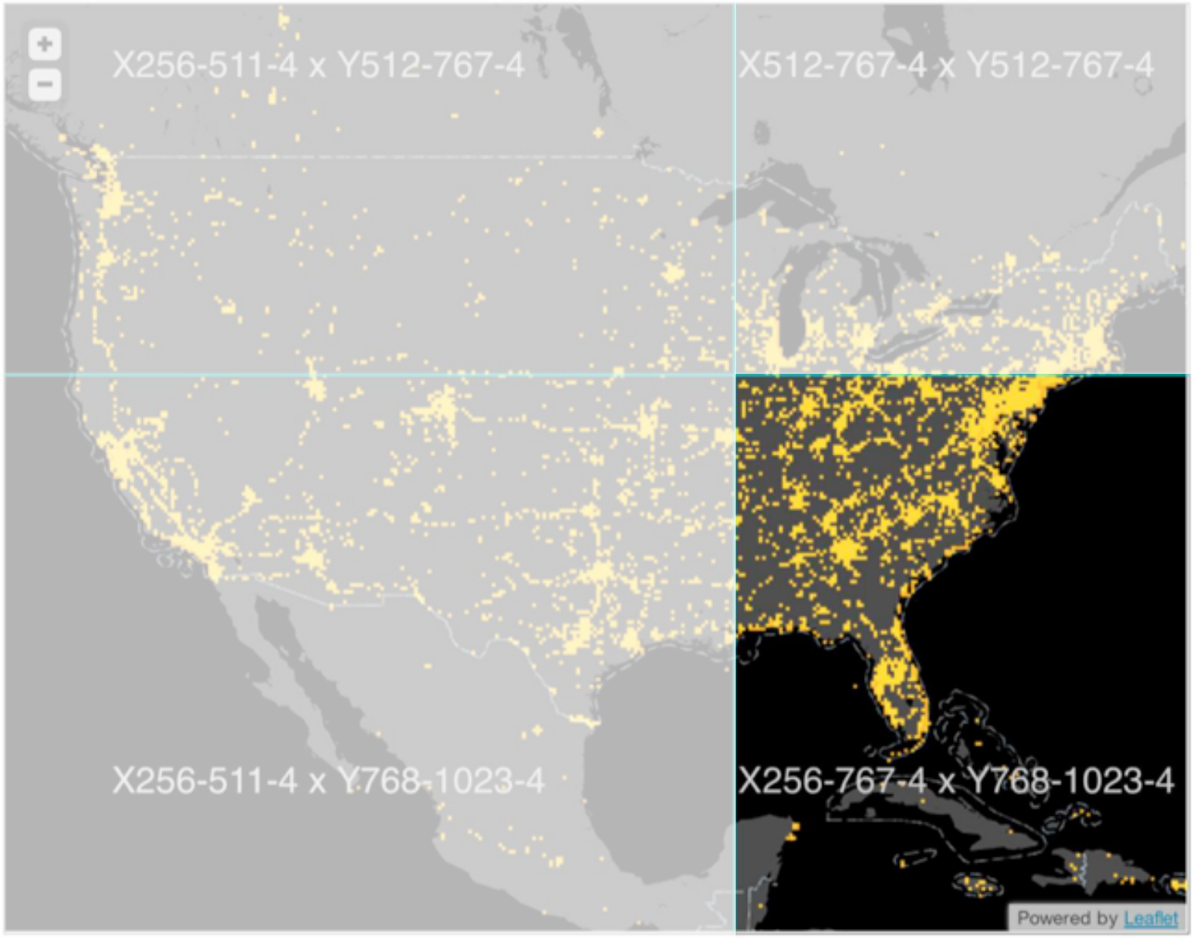

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

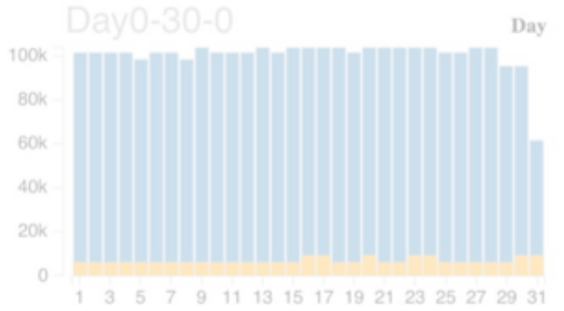
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

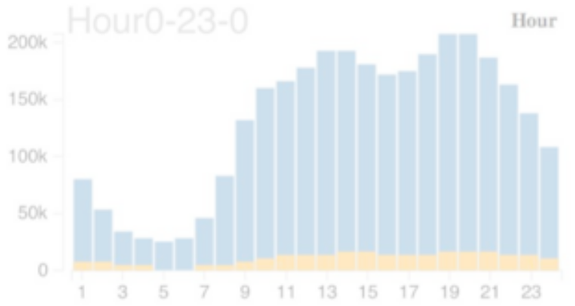

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

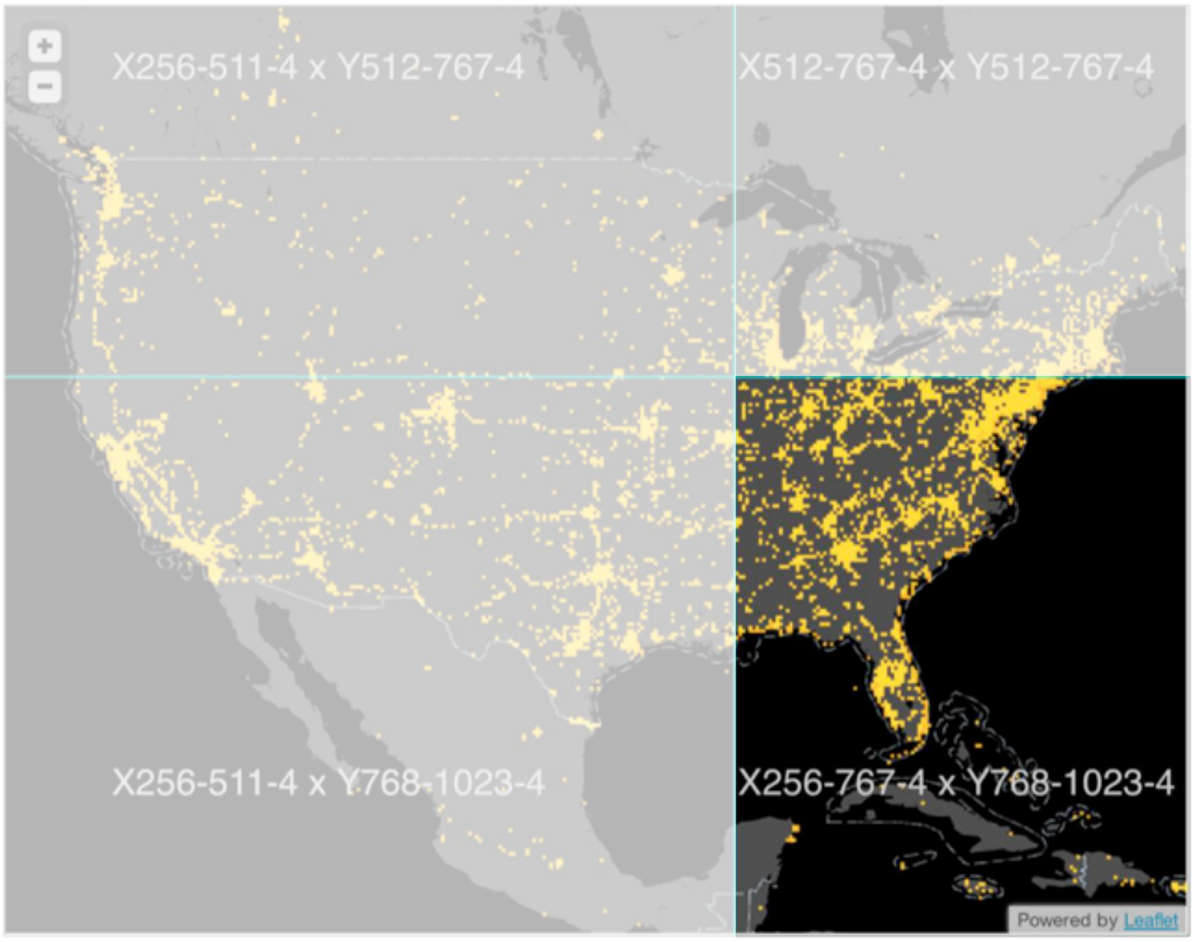

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

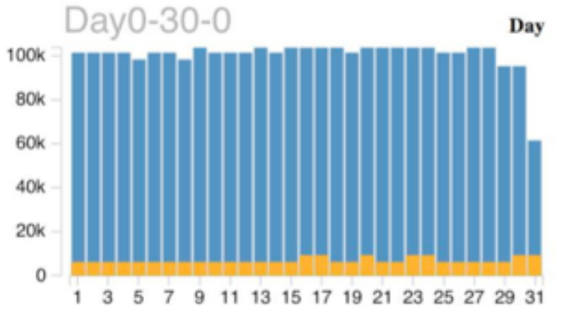
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

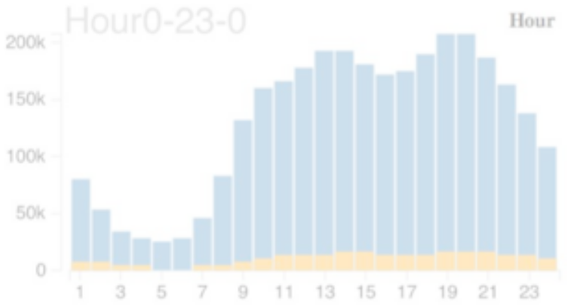

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

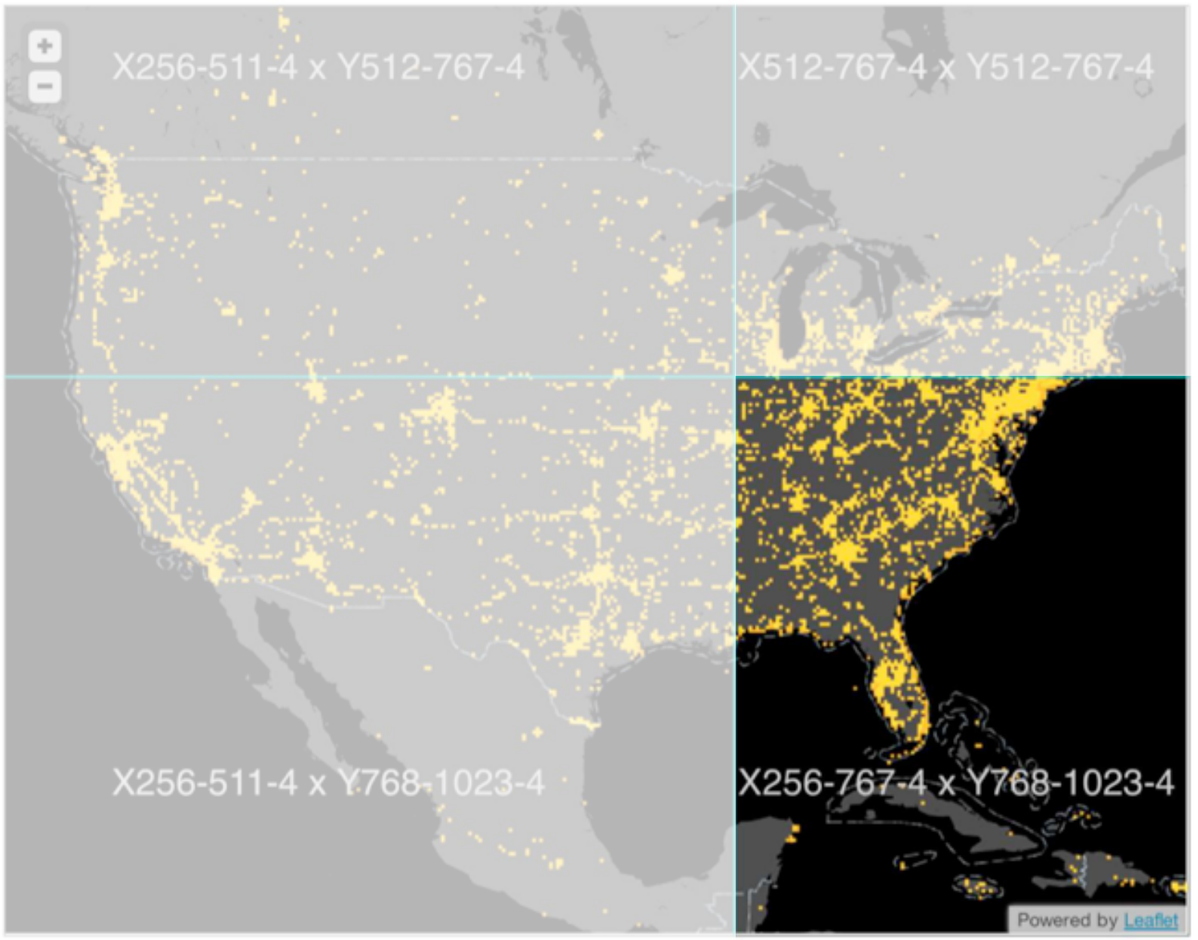

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

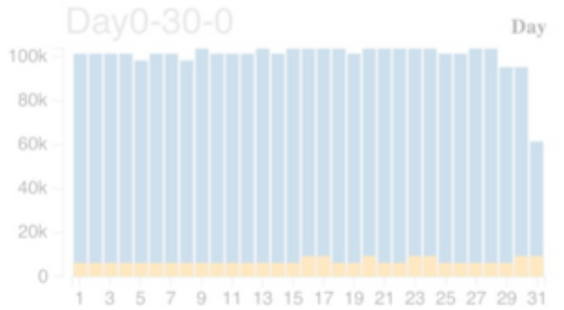
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

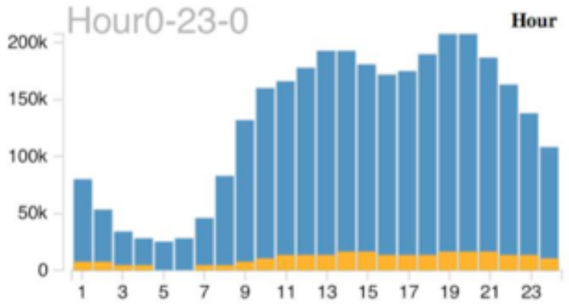

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

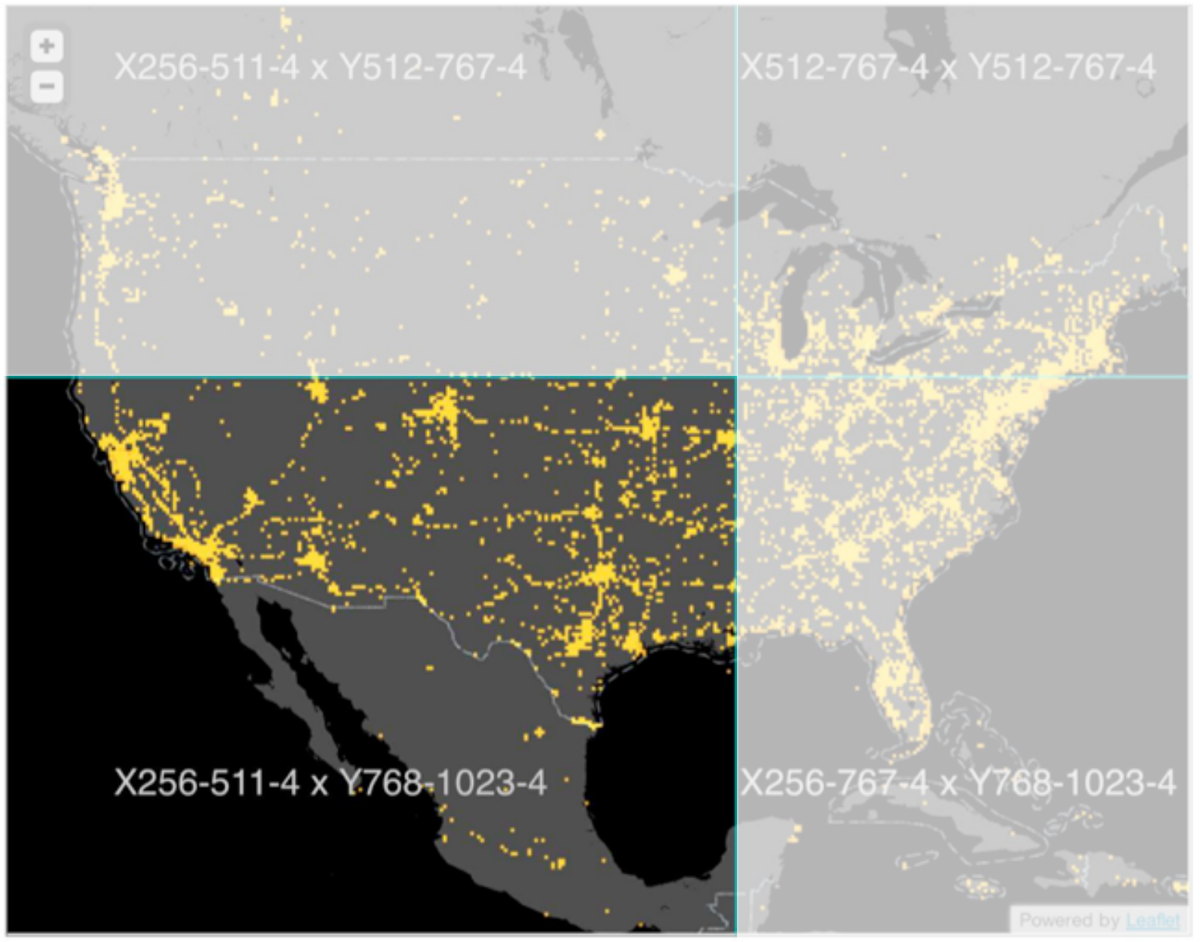

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

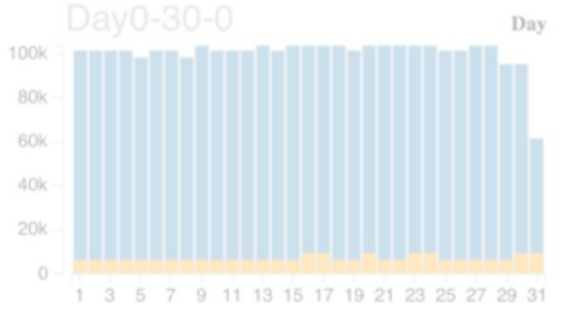
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

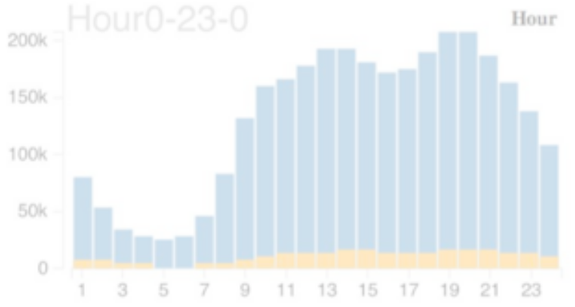

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

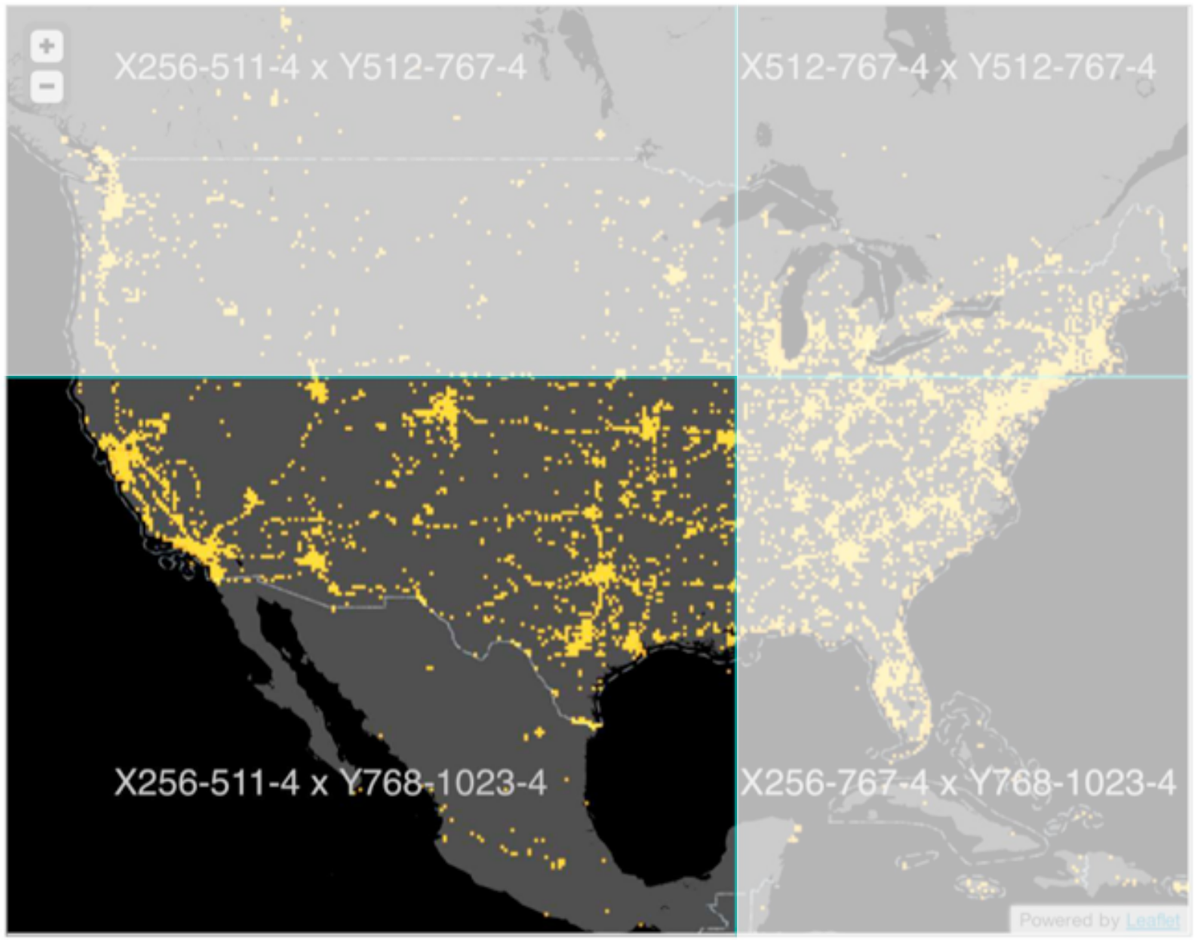

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

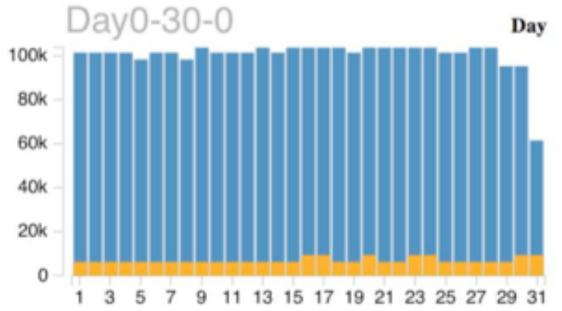
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

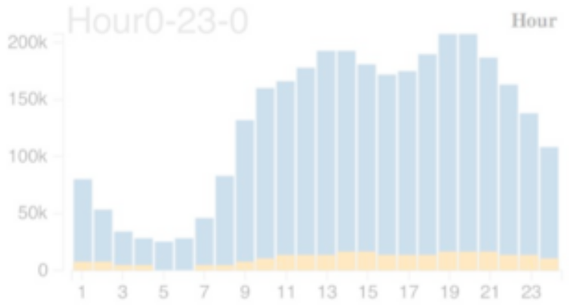

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

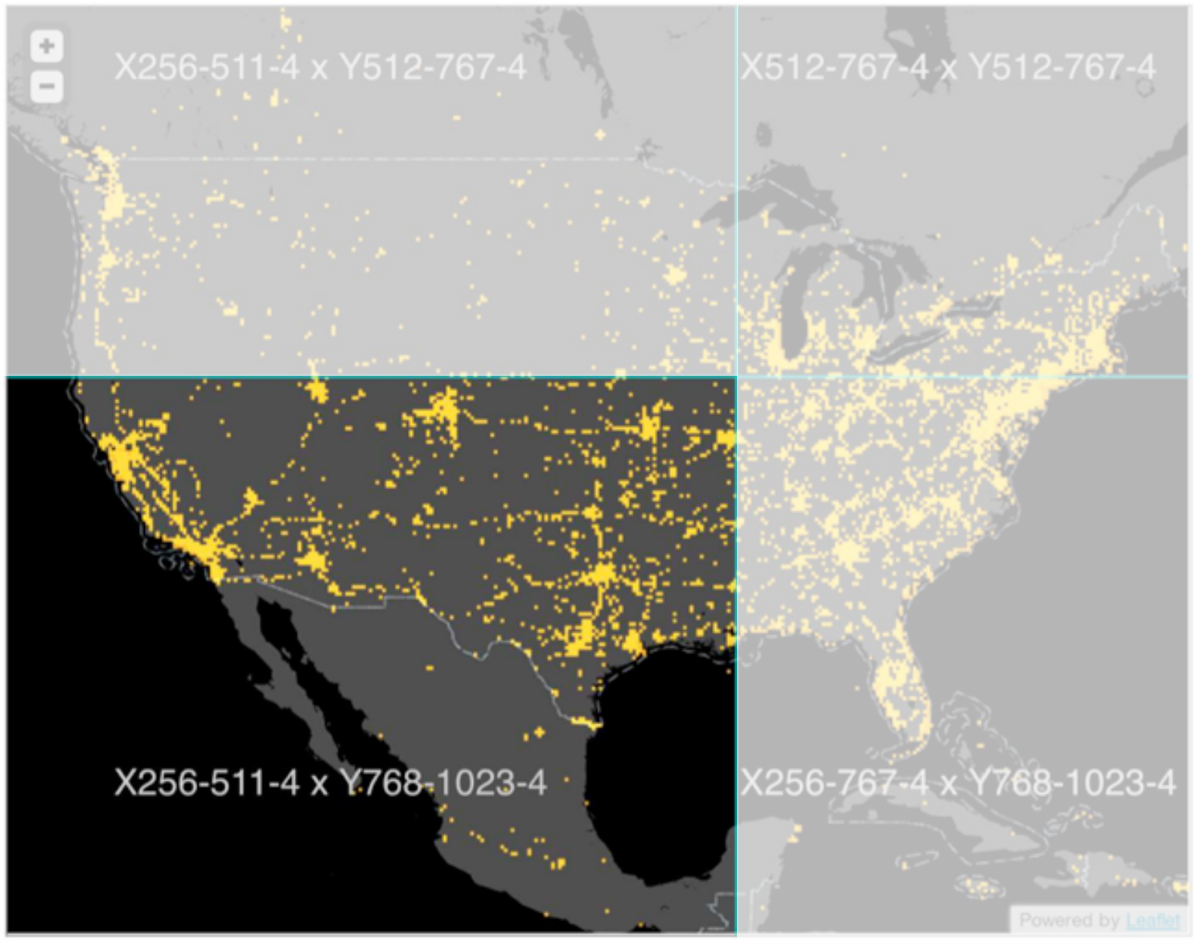

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

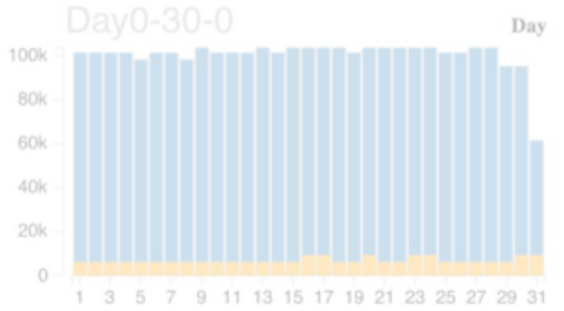
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

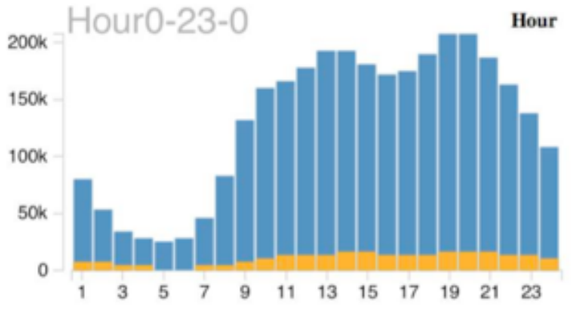

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

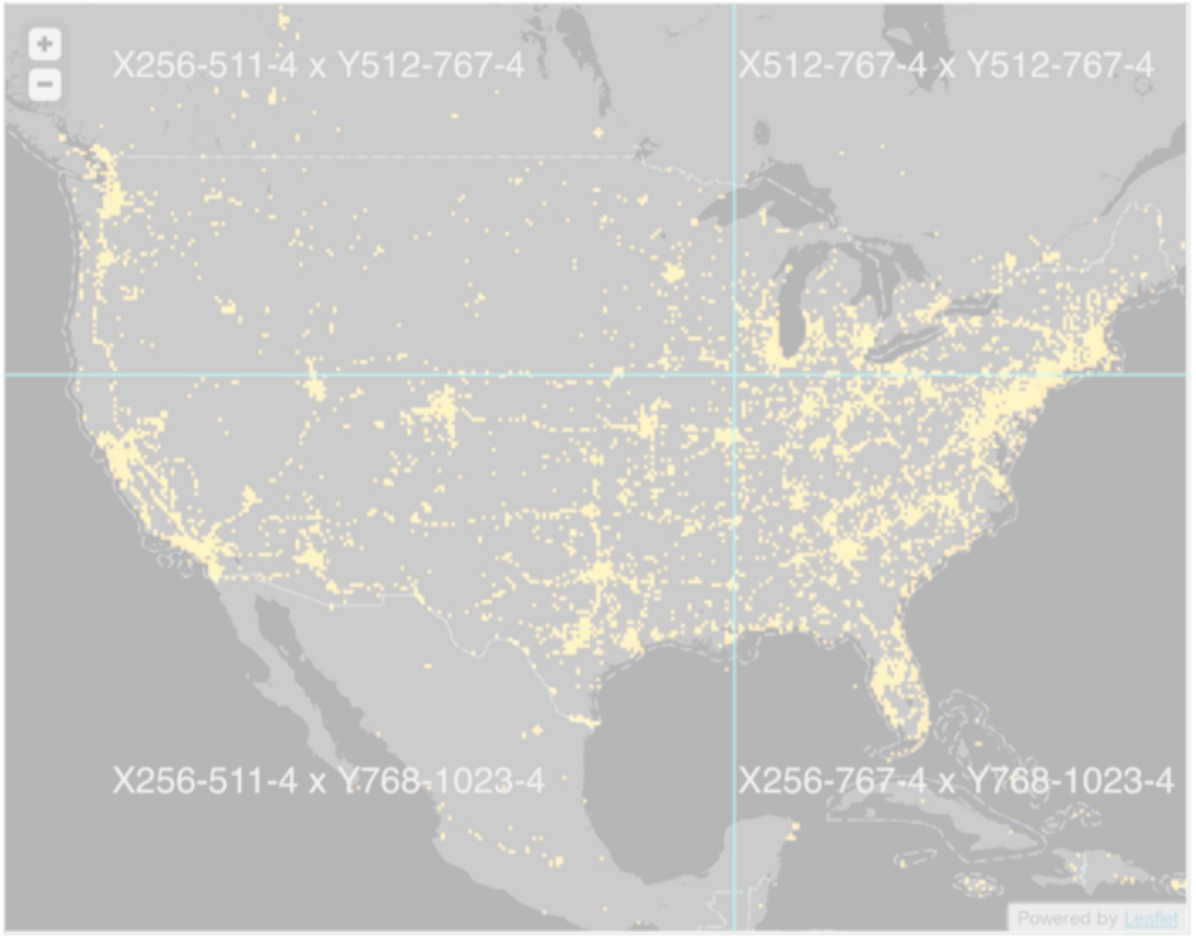

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

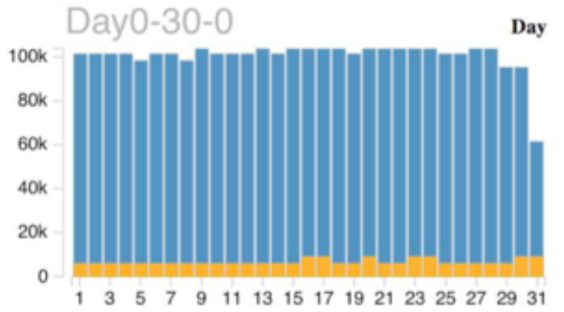
Month0-11-0
Month
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

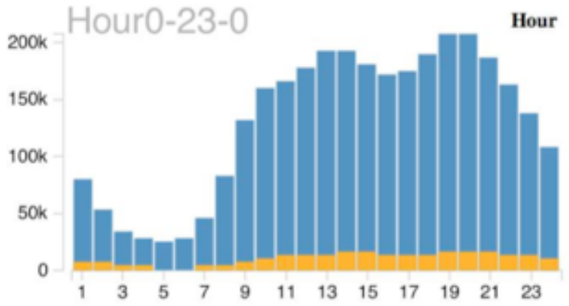

Day0-30-0
Day
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

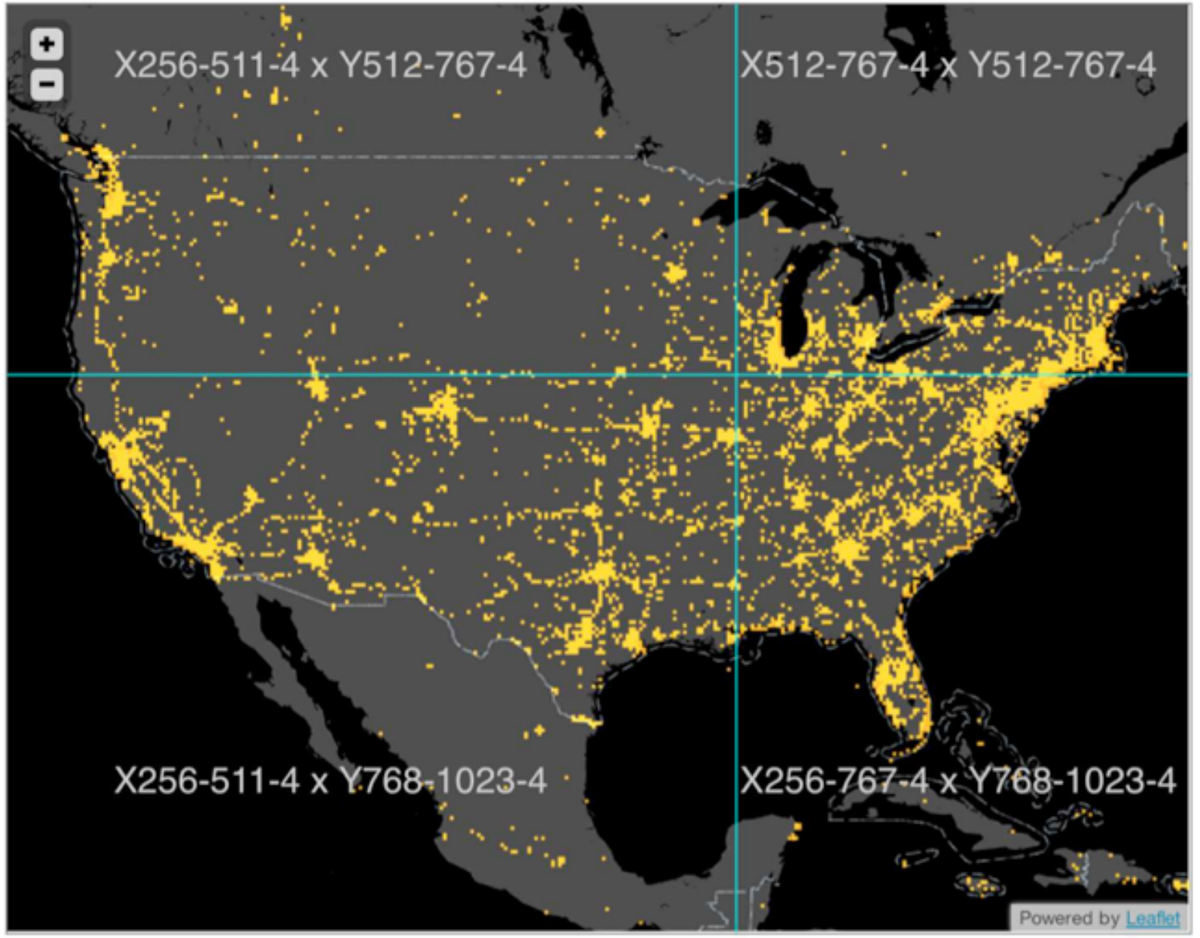

Hour0-23-0
Hour
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23


X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

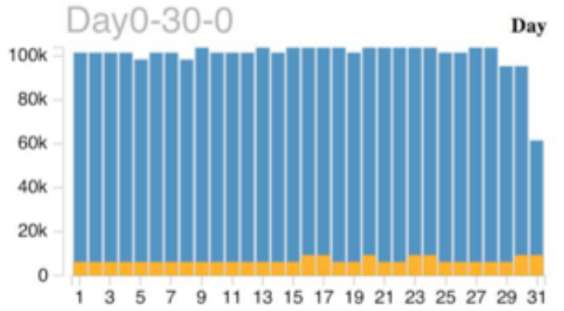
Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

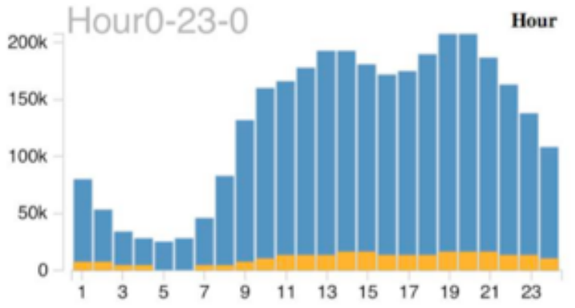

Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31


Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23

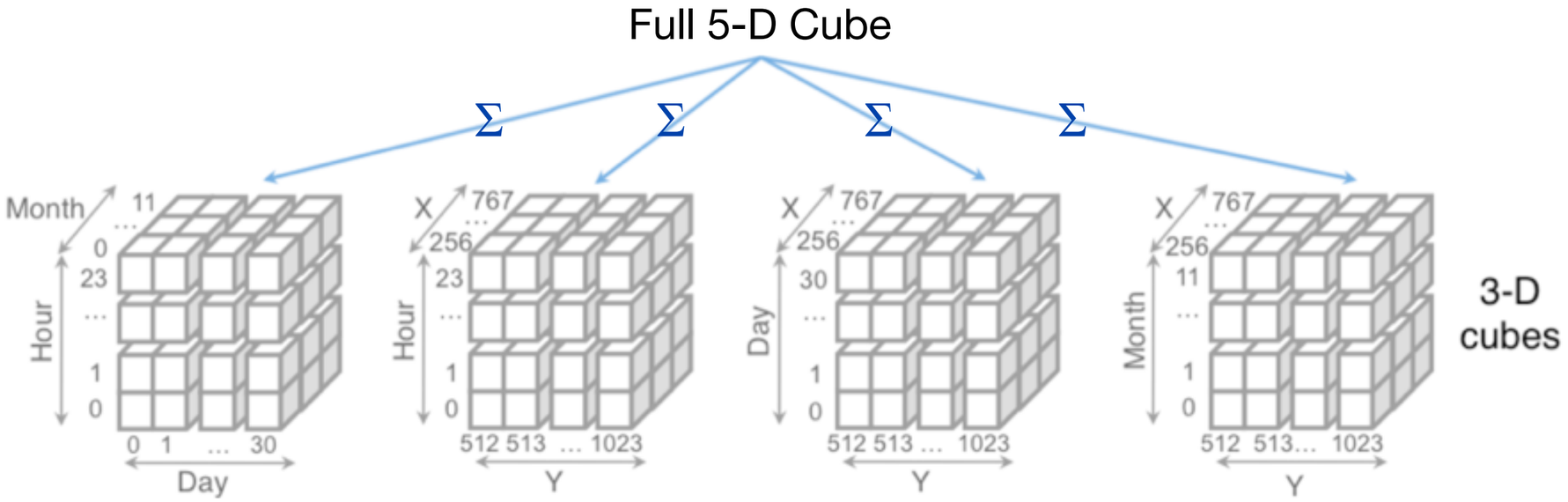

X256-511-4 x Y512-767-4
X512-767-4 x Y512-767-4
X256-511-4 x Y768-1023-4
X256-767-4 x Y768-1023-4
Powered by Leaflet

Month
Month0-11-0
300k
250k
200k
150k
100k
50k
0
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec


Day
Day0-30-0
100k
80k
60k
40k
20k
0
1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31

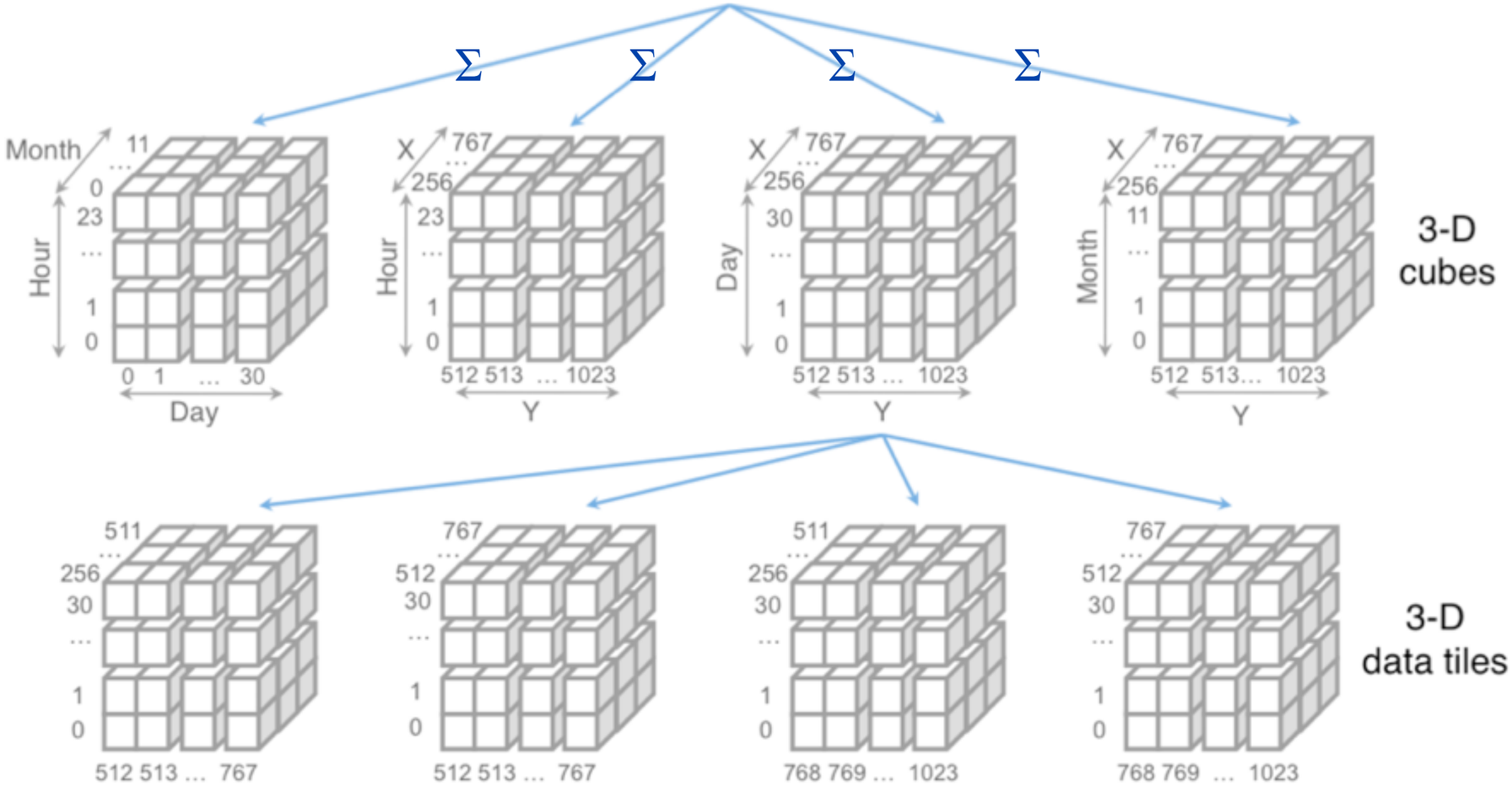

Hour
Hour0-23-0
200k
150k
100k
50k
0
1 3 5 7 9 11 13 15 17 19 21 23

Full 5-D Cube

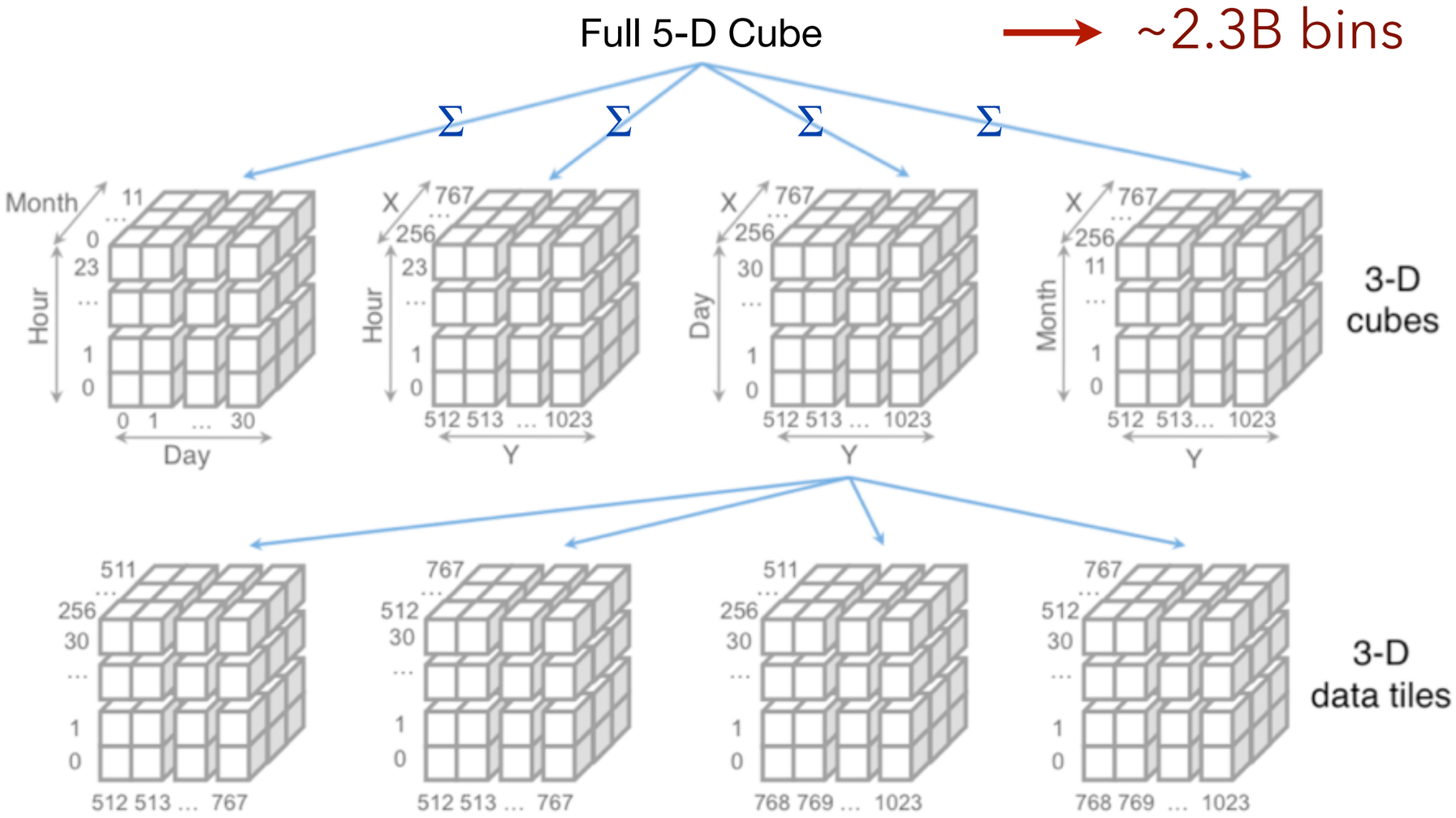


For any pair of 1D or 2D binned plots, the maximum number of dimensions needed to support brushing & linking is **four**.

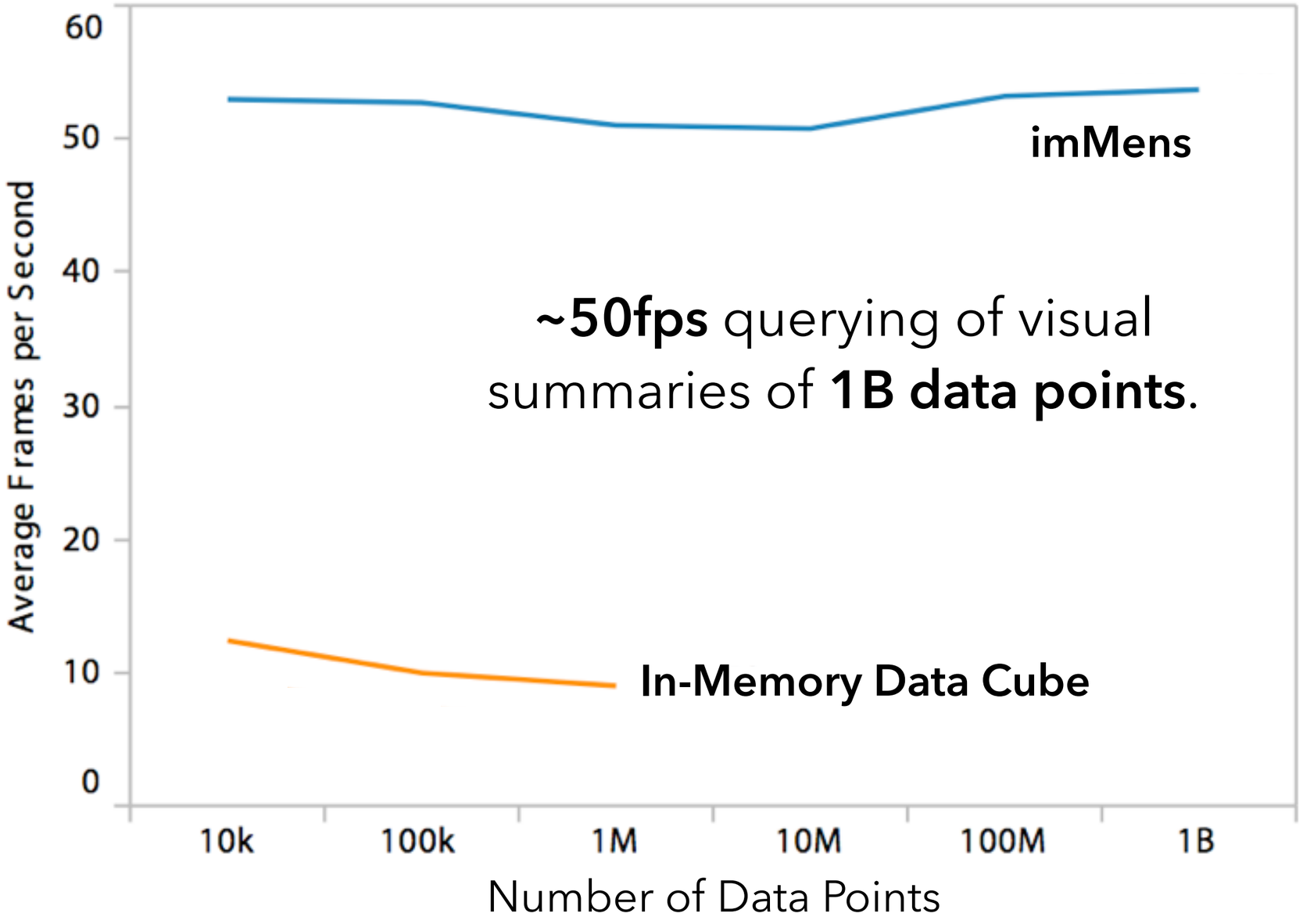
Full 5-D Cube



13 3-D Data Tiles


Full 5-D Cube
~2.3B bins
Σ
Σ
Σ
Σ
Month
11
0
Hour
23
1
0
0 1 ... 30
Day
X
767
256
Hour
23
1
0
512 513 ... 1023
Y
X
767
256
Day
30
1
0
512 513 ... 1023
Y
X
767
256
Month
11
1
0
512 513... 1023
Y
3-D cubes
511
256
30
1
0
512 513 ... 767
767
512
30
1
0
512 513 ... 767
511
256
30
1
0
768 769 ... 1023
767
512
30
1
0
768 769 ... 1023
3-D data tiles
13 3-D Data Tiles
~17.6M bins
(in 352KB!)

5 dimensions x 50 bins/dim x 25 plots



Limitations and Questions

But where do the multivariate data tiles come from?

They must be provided by a backend server. This can be time-consuming, particularly if supporting deep levels of zooming. imMens assumes that tiles have either been pre-computed or that a backing database can suitably generate them on demand.

Does super-low-latency interaction really matter?

Is it worth it to go to all of this trouble? (Short answer: yes!)

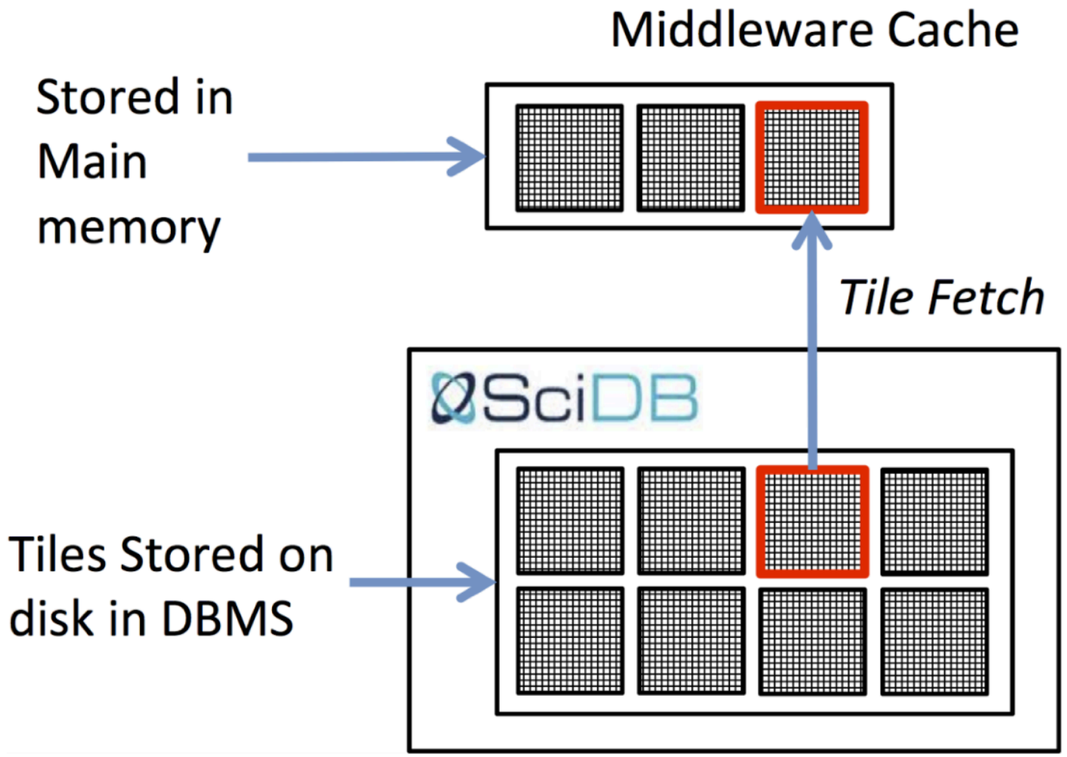
High latency leads to reduced analytic output [Liu & Heer, InfoVis 2014]

ForeCache

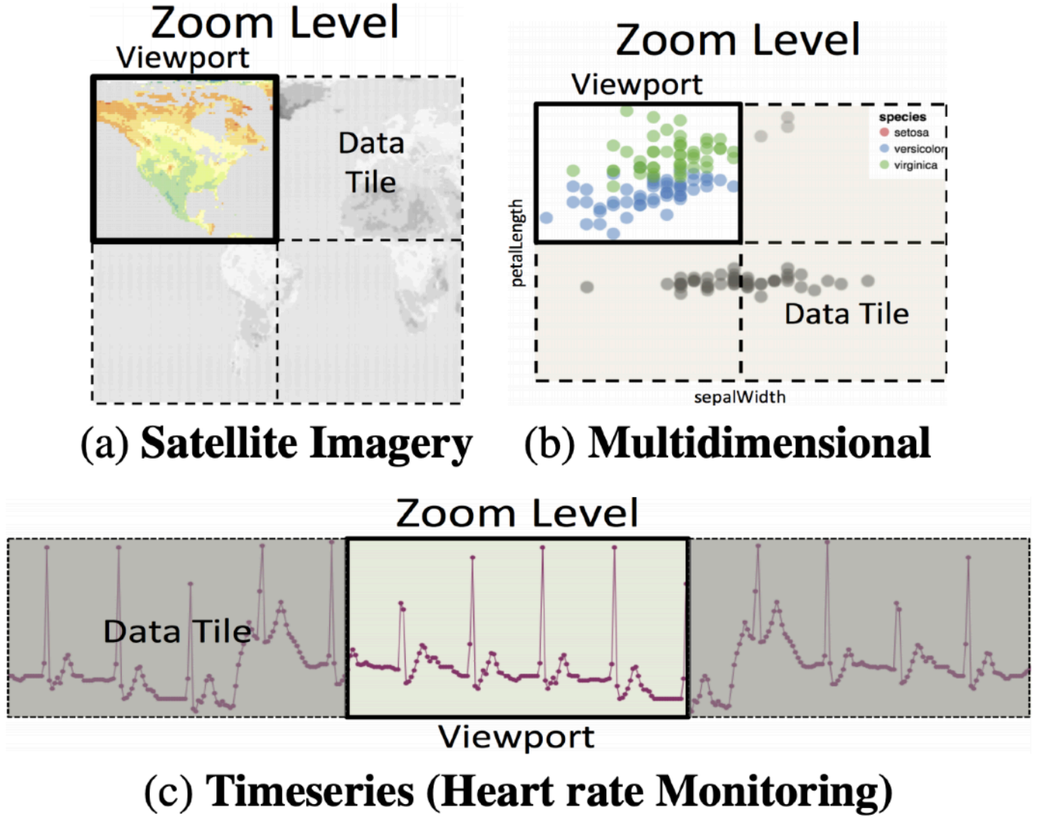
[Battle, Chang, & Stonebraker '16]

Strategies: Query Database, Prefetching

ForeCache is also a Data Tile-Based System



Manage a Cache of Tiles from DB



Example Tile-Based Views

Key Idea: Model & Predict User Behavior

1. Classify Analysis Phase

Foraging: Searching for patterns of interest
Sensemaking: Closely examine a region-of-interest (ROI)
Navigation: Transition between levels of detail

Train a machine learning classifier (SVM) to predict phase.
The input data is the activity trace of user interactions.

Key Idea: Model & Predict User Behavior

1. Classify Analysis Phase

2. Apply Prediction Models

Actions-Based: Use recent interactions to predict next ones.

You pan left twice; what is the probability you will do it again?

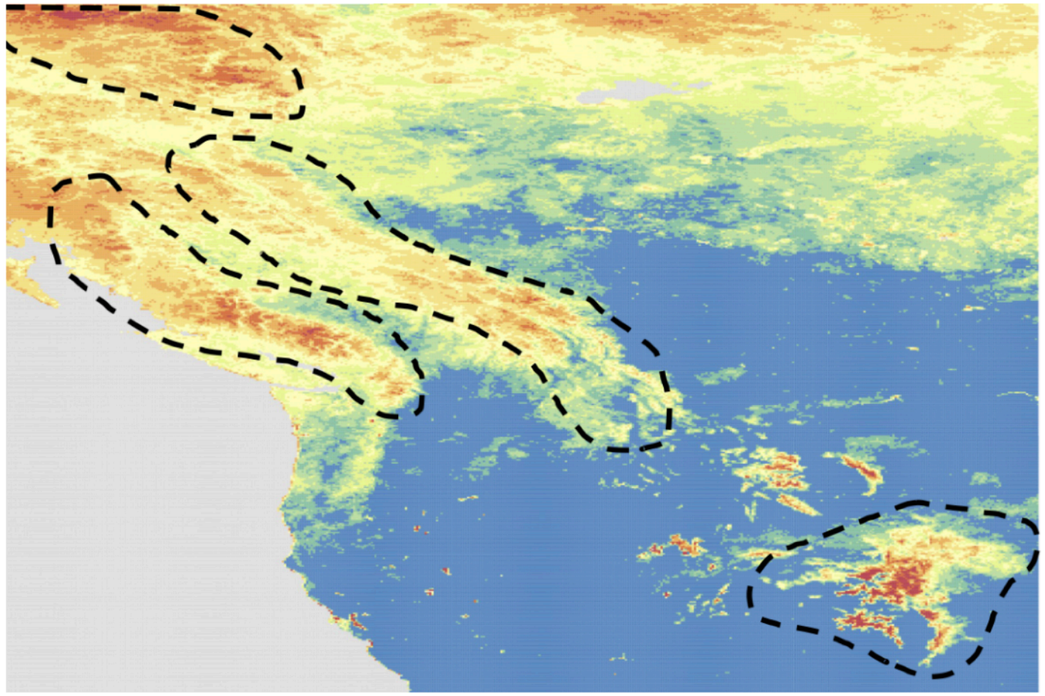
Signature-Based: Match to data characteristics of interest.

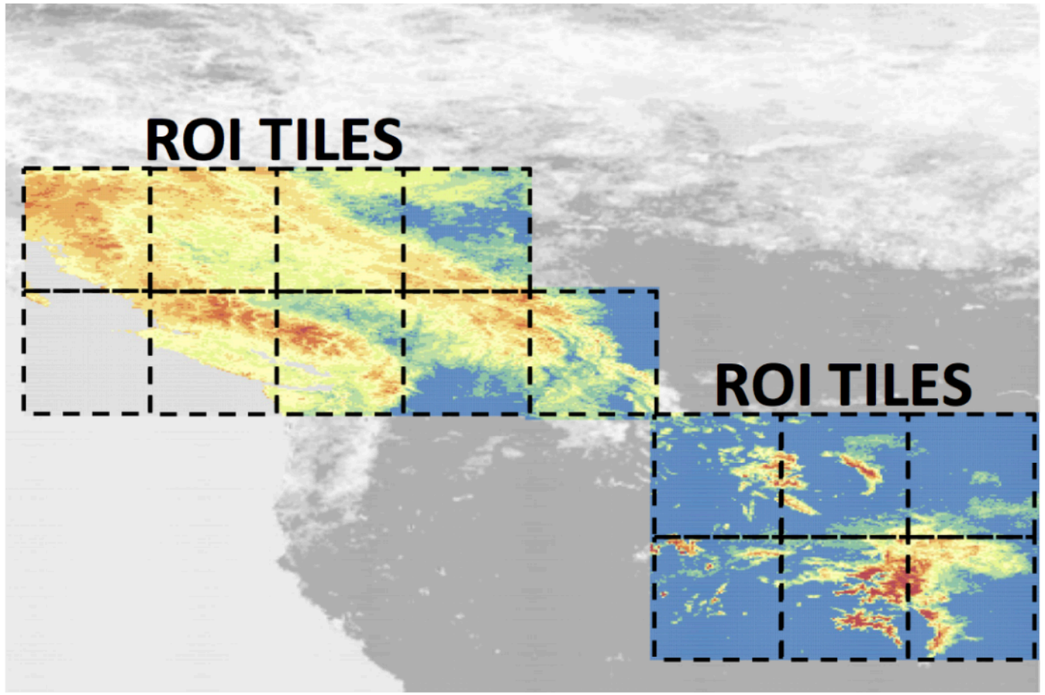
What data tiles are visually similar to current focus tiles?

These models are weighted based on the analysis phase.

Actions-Based for *navigation*. Signature-Based for *sensemaking*. Both applied equally for *foraging*.

Application: MODIS Satellite Data





Analyzing snow cover in a scientific database. ROI = Region of Interest

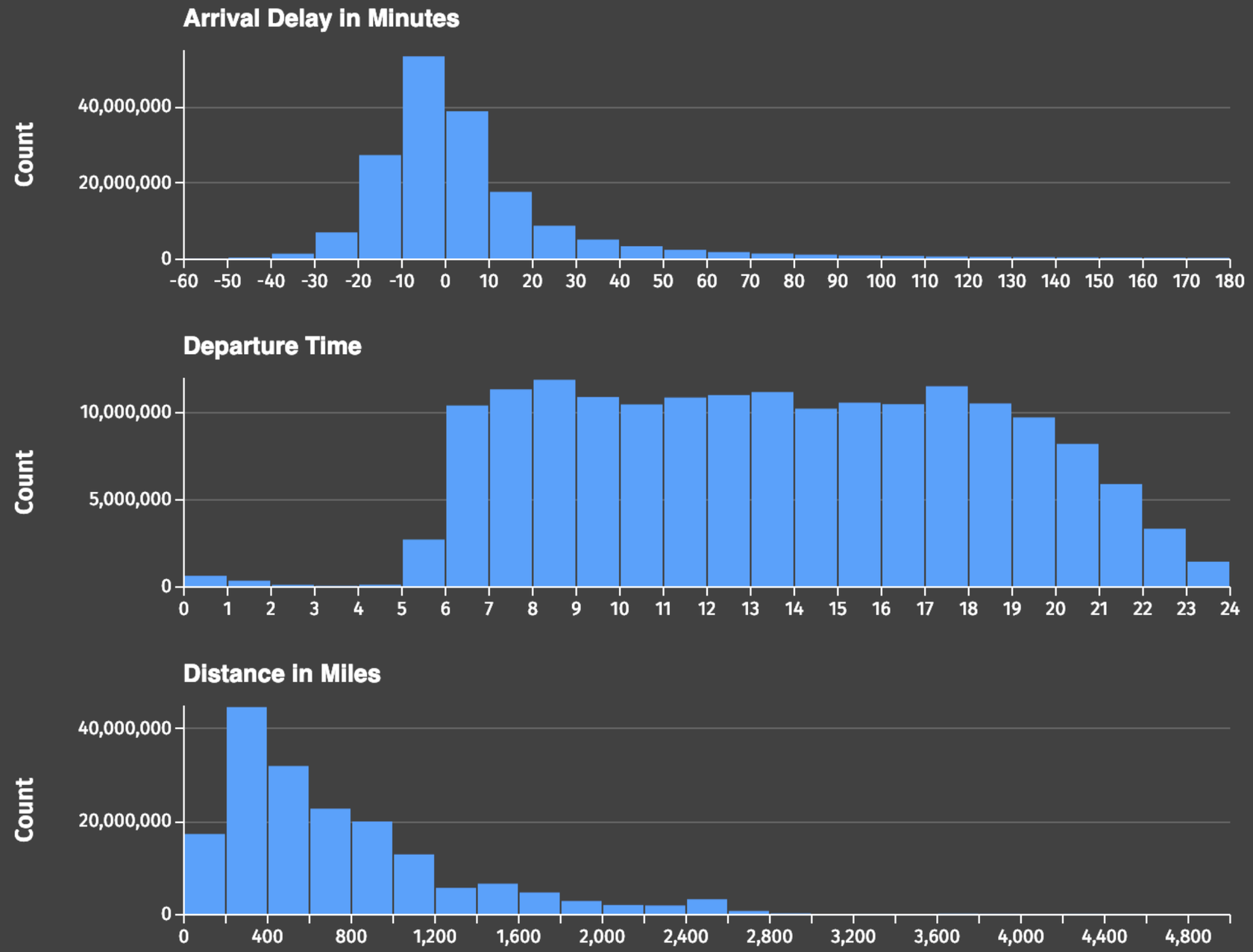
ForeCache improves latency:

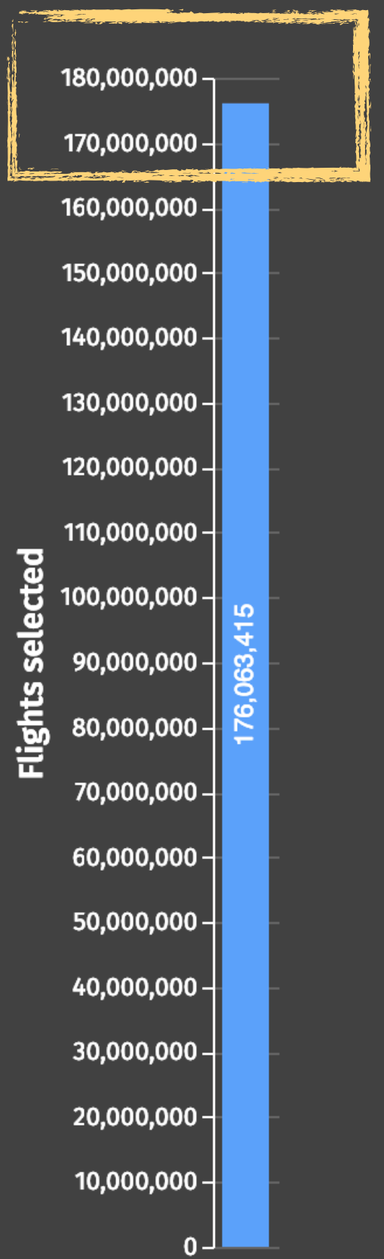
- 430% better than current non-prefetching systems
- 88% better than existing prediction methods

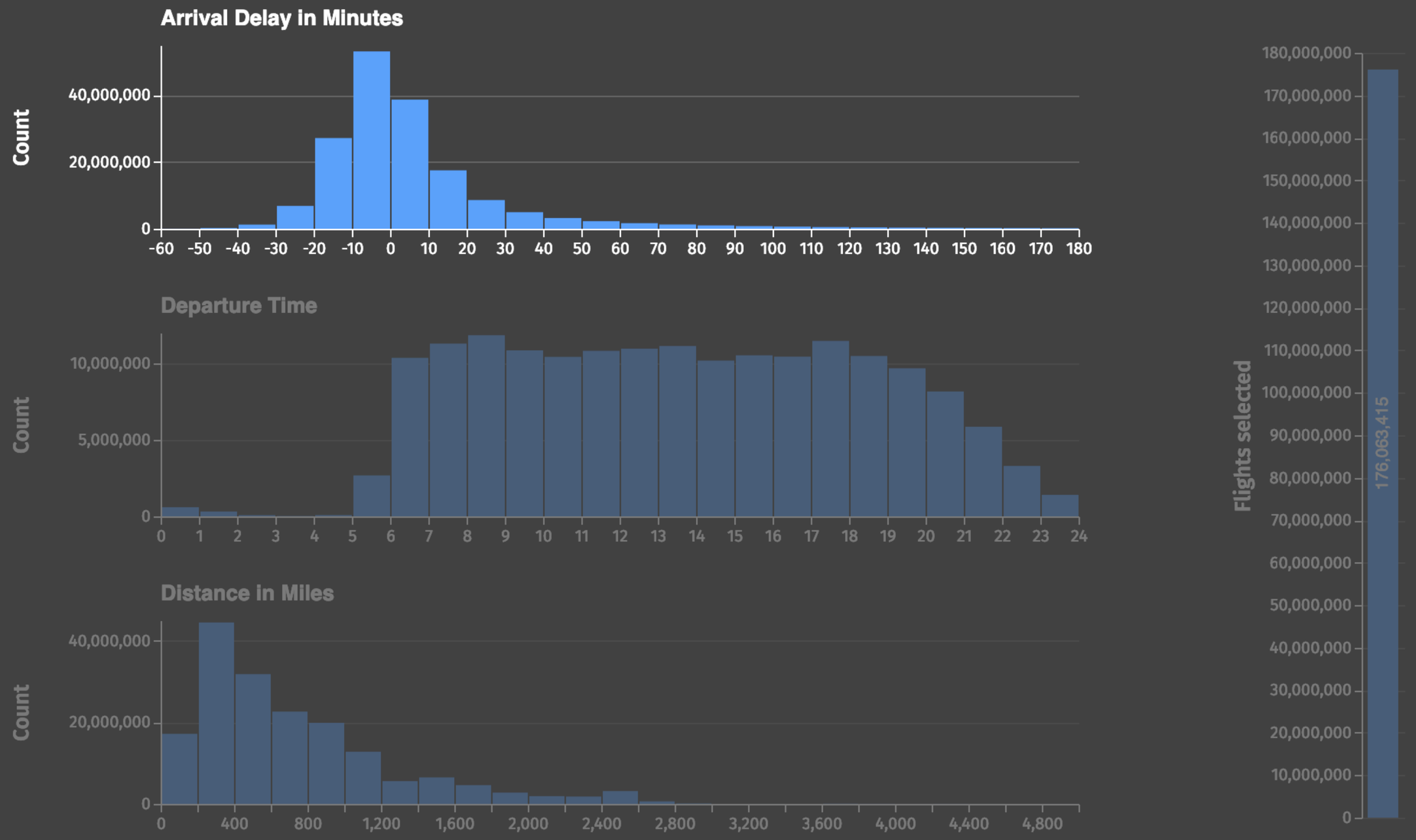
Falcon

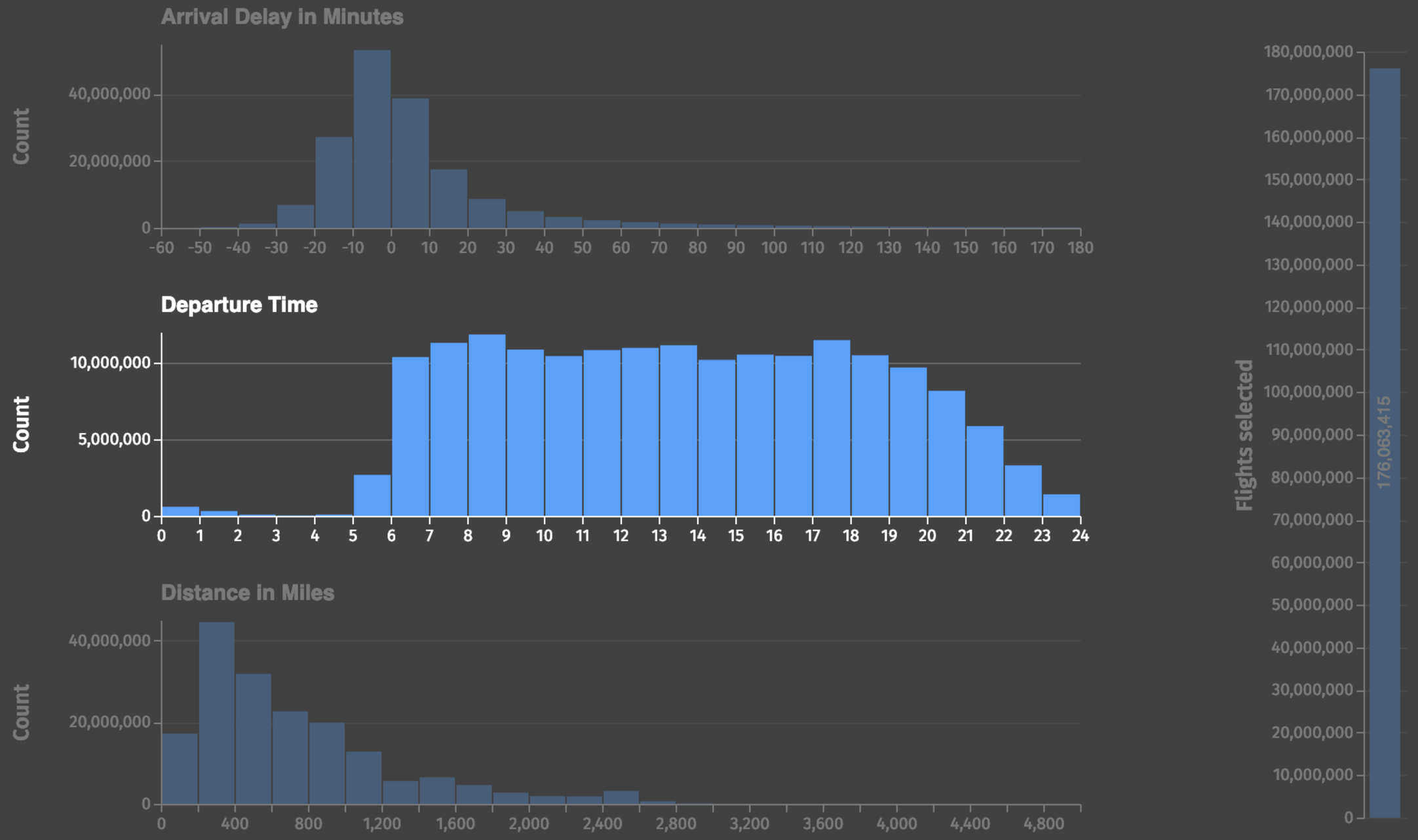
[Moritz, Howe, & Heer ’19]

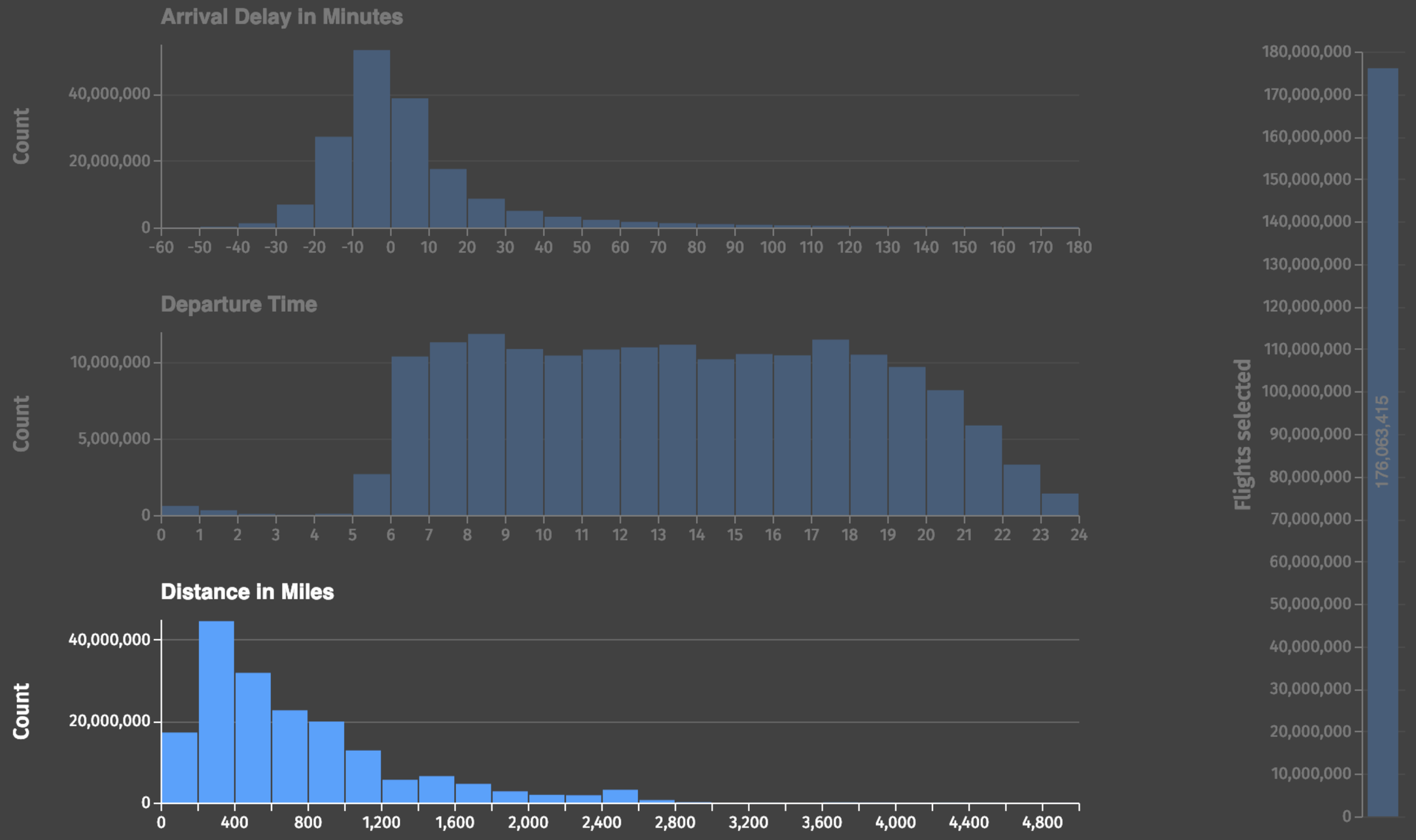
Strategies: Query Database, Client-Side Data Cubes, Prefetching

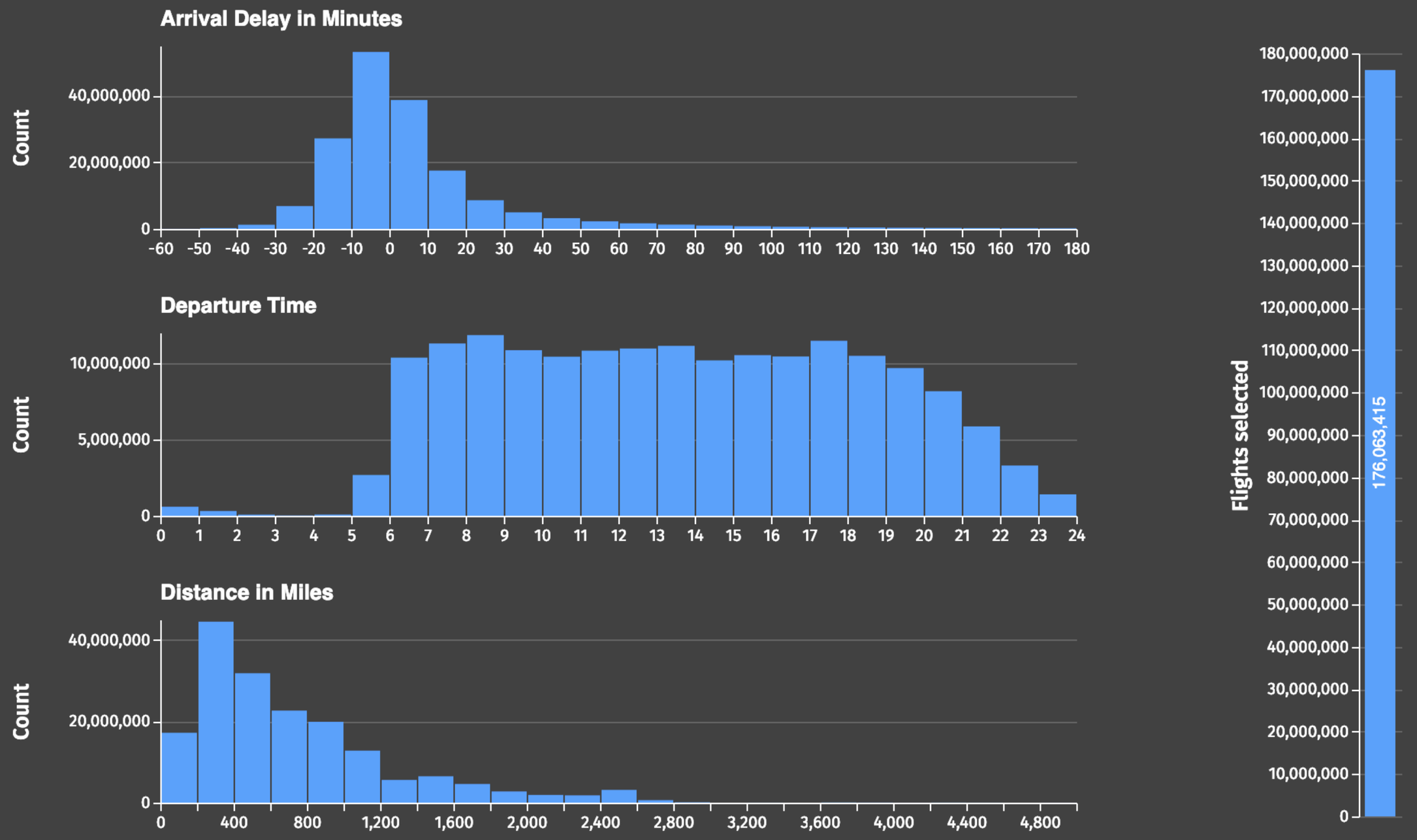

Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800

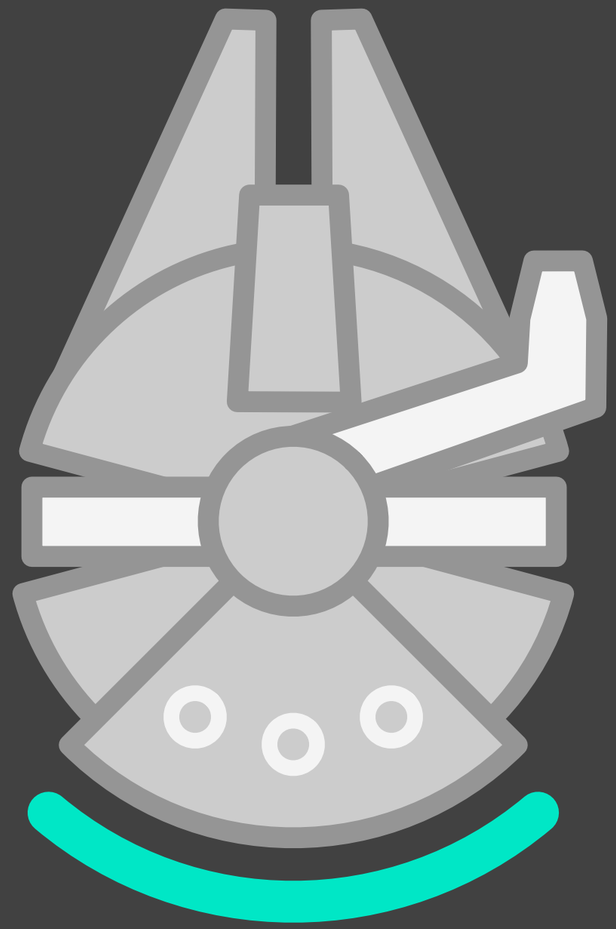

Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415


Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415


Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415

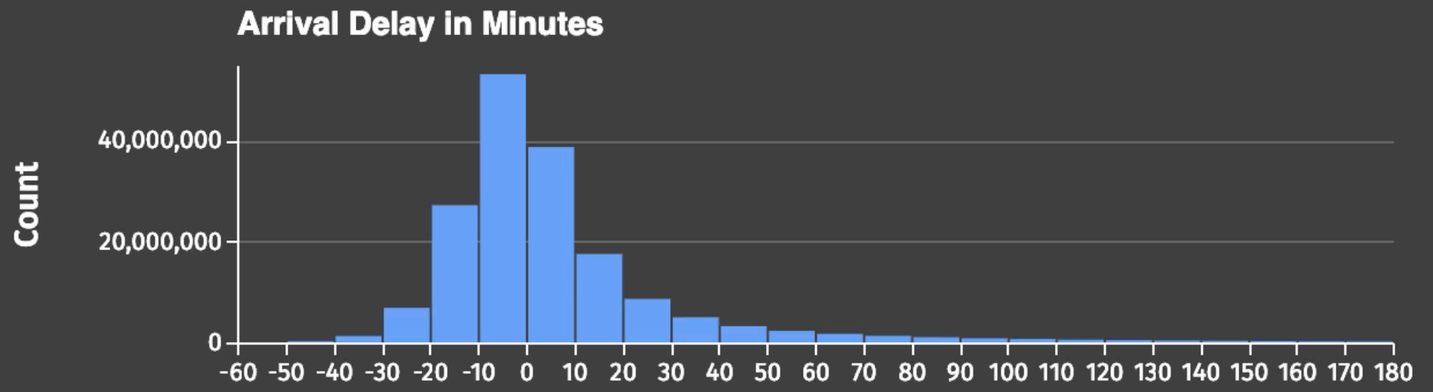

Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415

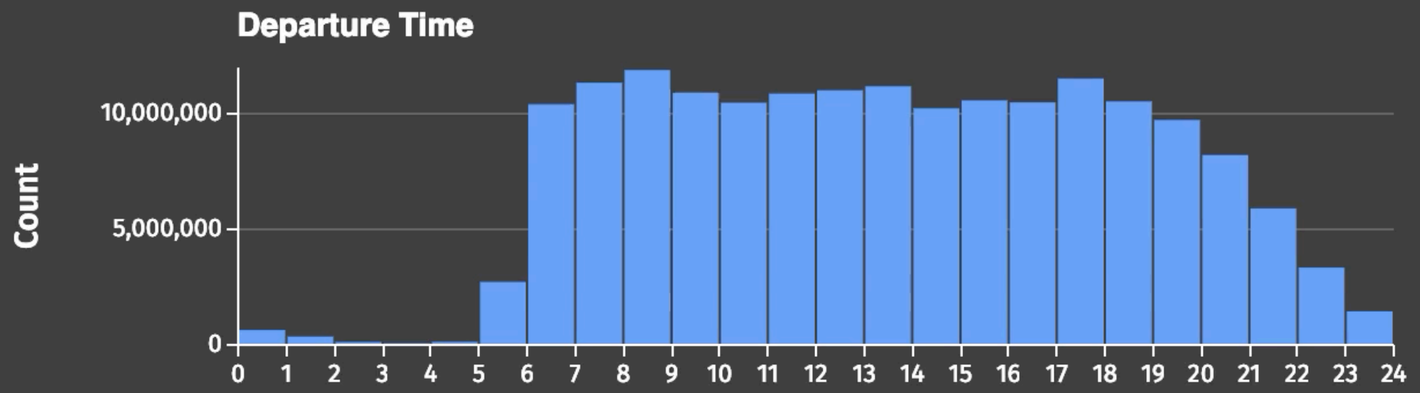

Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415

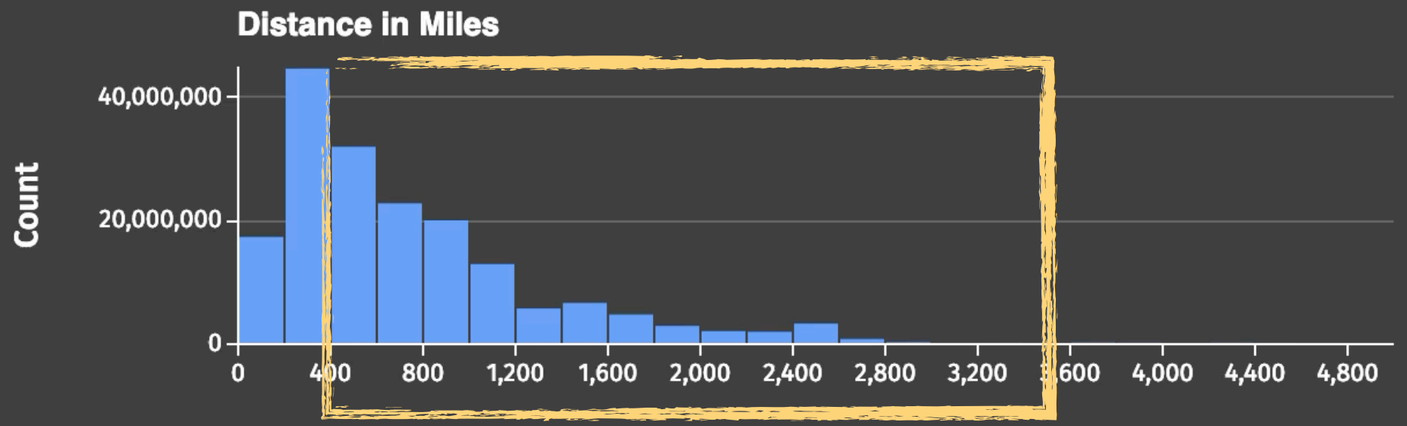


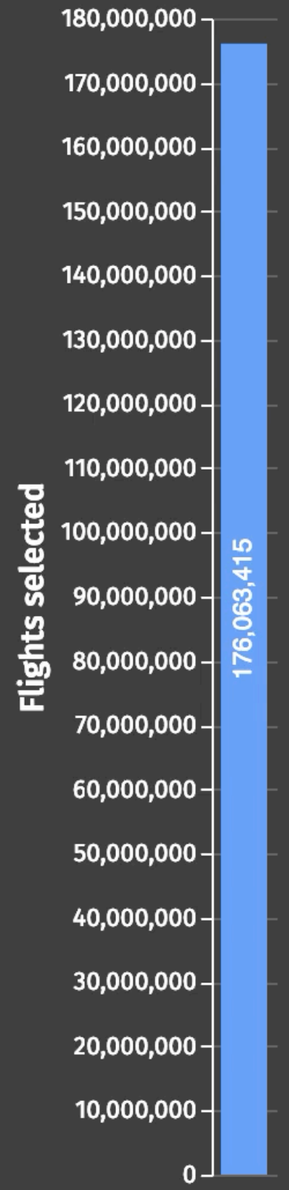
Falcon

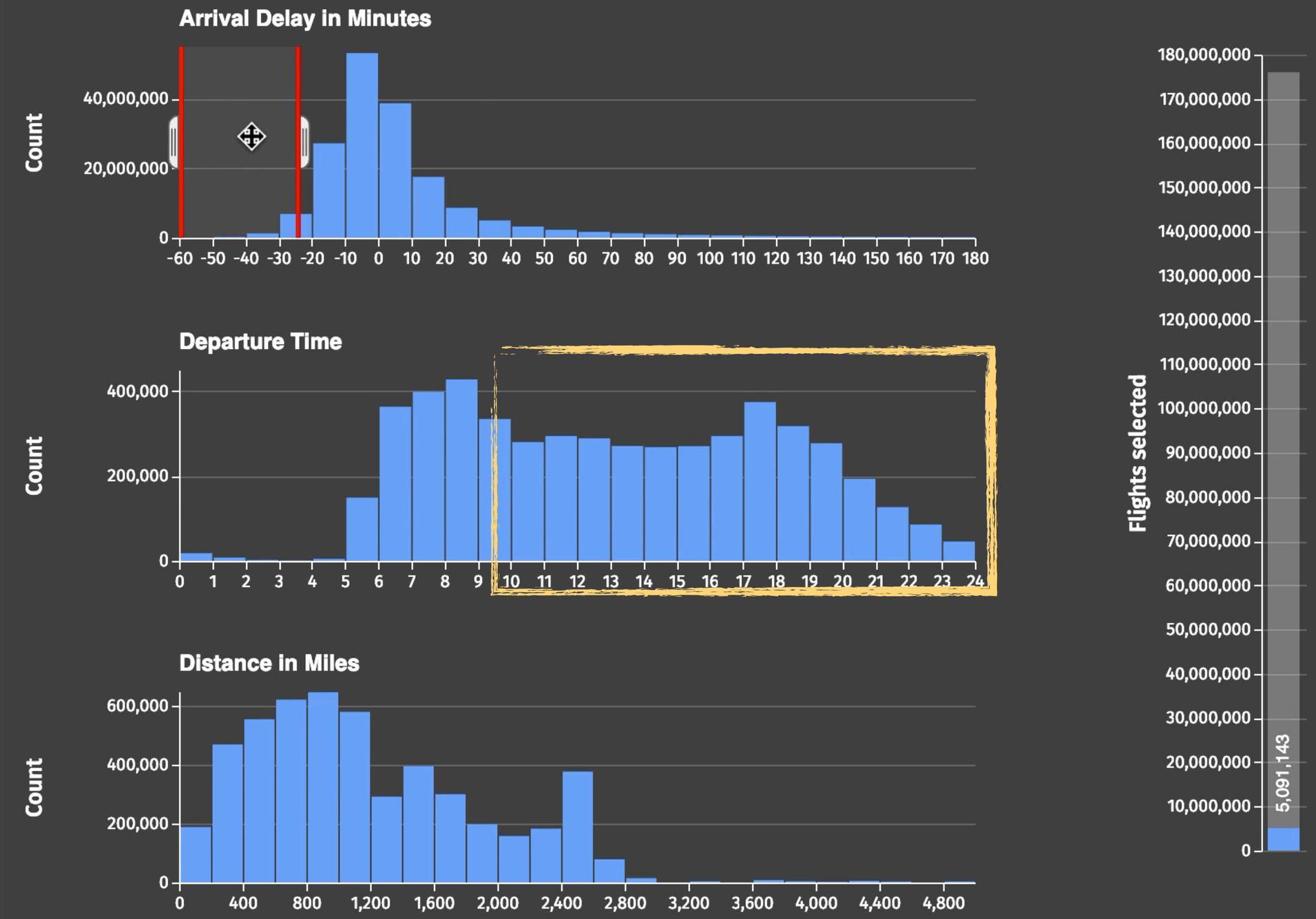
uwdata.github.io/falcon

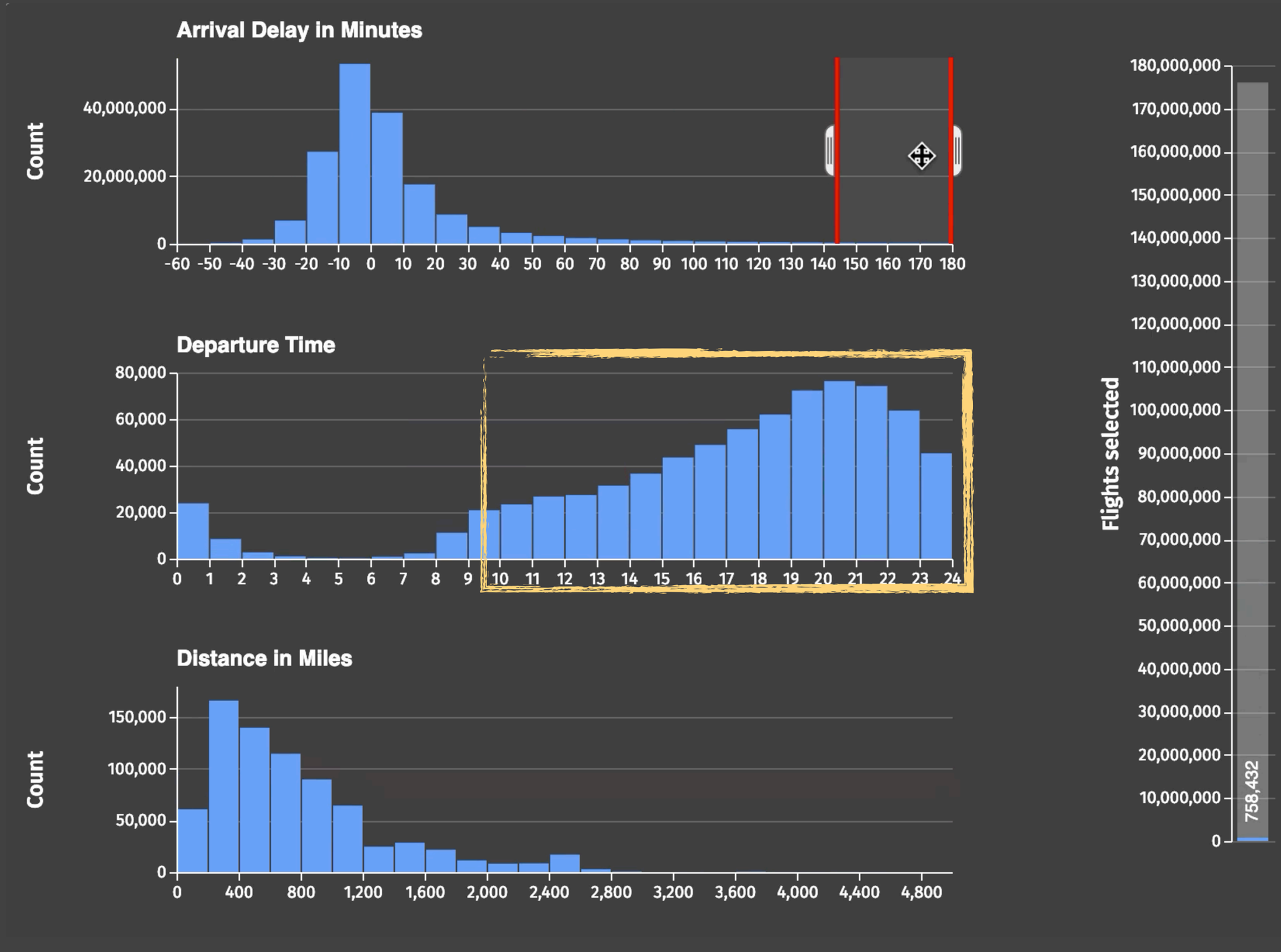

Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180

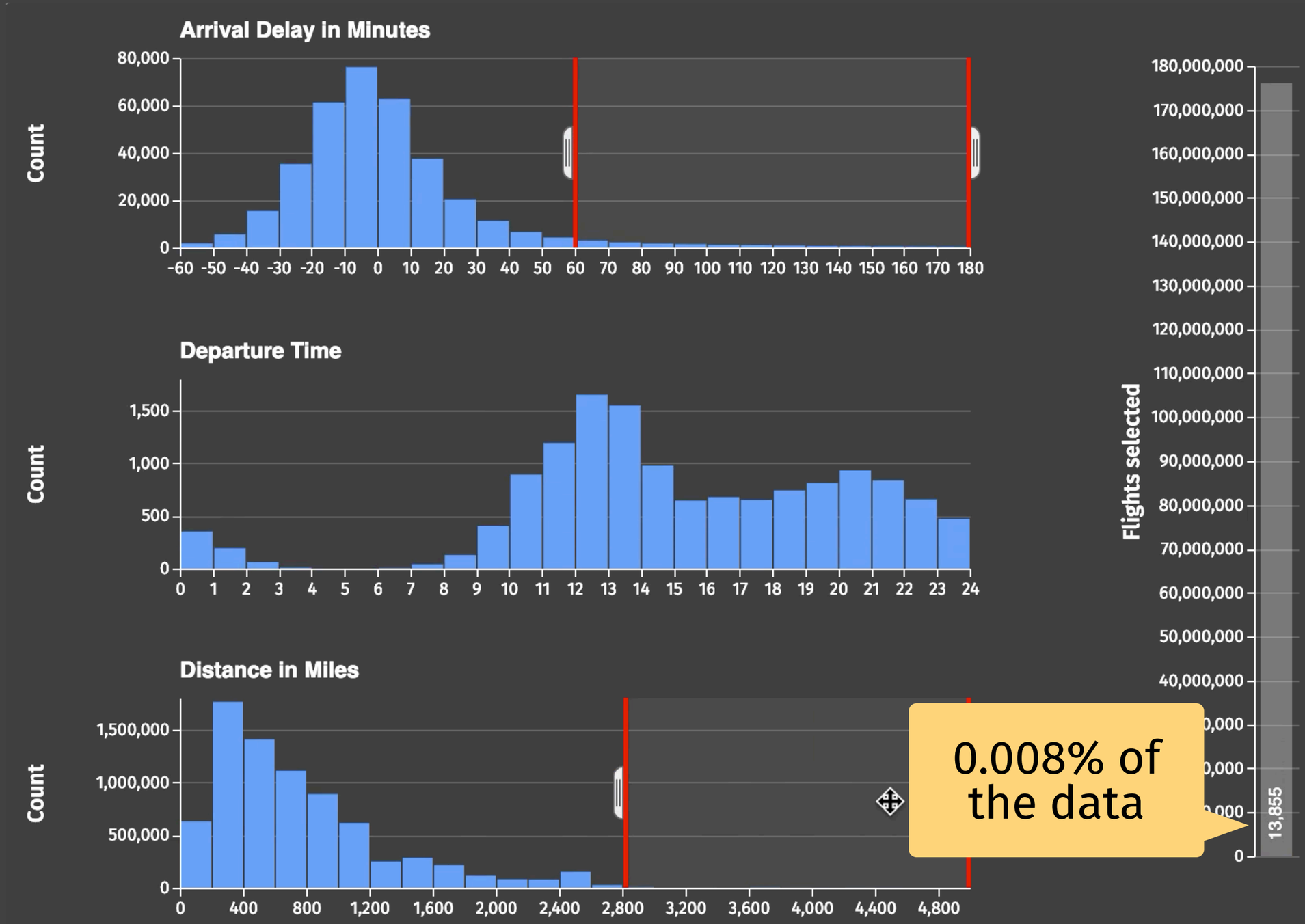

Departure Time
Count
10,000,000
5,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24


Distance in Miles
Count
40,000,000
20,000,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800


Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
176,063,415


Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
400,000
200,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
600,000
400,000
200,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
5,091,143

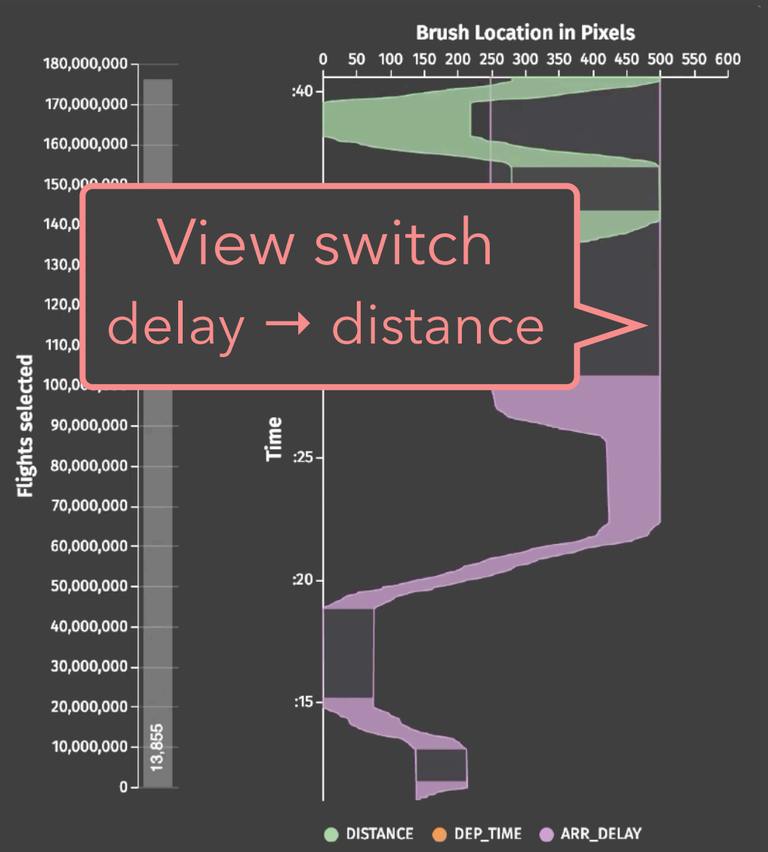

Arrival Delay in Minutes
Count
40,000,000
20,000,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
80,000
60,000
40,000
20,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
150,000
100,000
50,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
30,000,000
20,000,000
10,000,000
0
758,432


Arrival Delay in Minutes
Count
80,000
60,000
40,000
20,000
0
-60 -50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180
Departure Time
Count
1,500
1,000
500
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Distance in Miles
Count
1,500,000
1,000,000
500,000
0
0 400 800 1,200 1,600 2,000 2,400 2,800 3,200 3,600 4,000 4,400 4,800
Flights selected
180,000,000
170,000,000
160,000,000
150,000,000
140,000,000
130,000,000
120,000,000
110,000,000
100,000,000
90,000,000
80,000,000
70,000,000
60,000,000
50,000,000
40,000,000
0
13,855
0.008% of the data

How does Falcon support fine-grained real-time interaction?

Falcon Interaction Log

5x speedup



View switch
delay → distance

Brushing
interactions

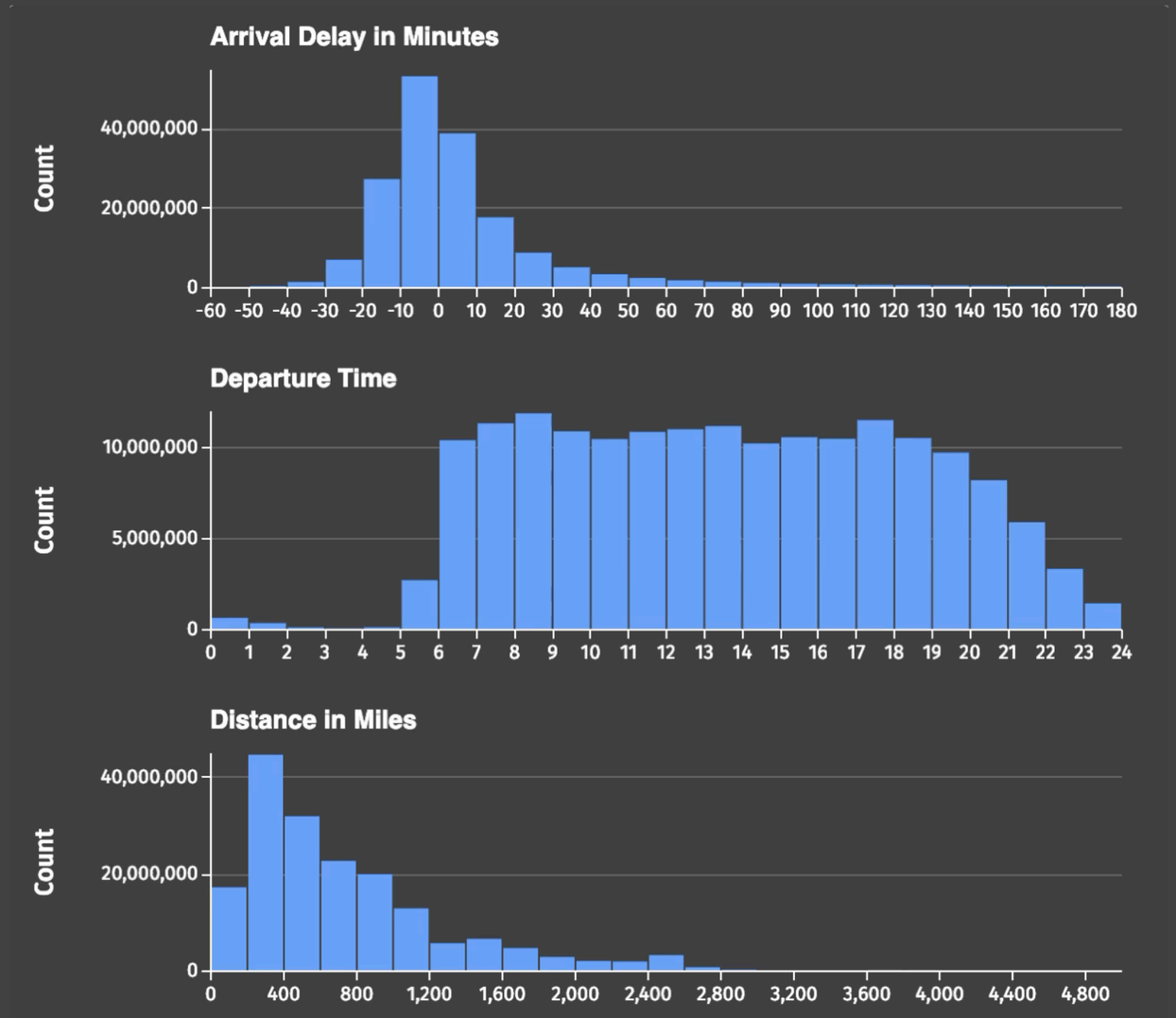
👁️ Brushing is more common and people are sensitive to latencies.

💡 Prioritize brushing latency over view switching latency.

Key Idea:

User-centered prefetching and indexing to support all brushing interactions with one view.

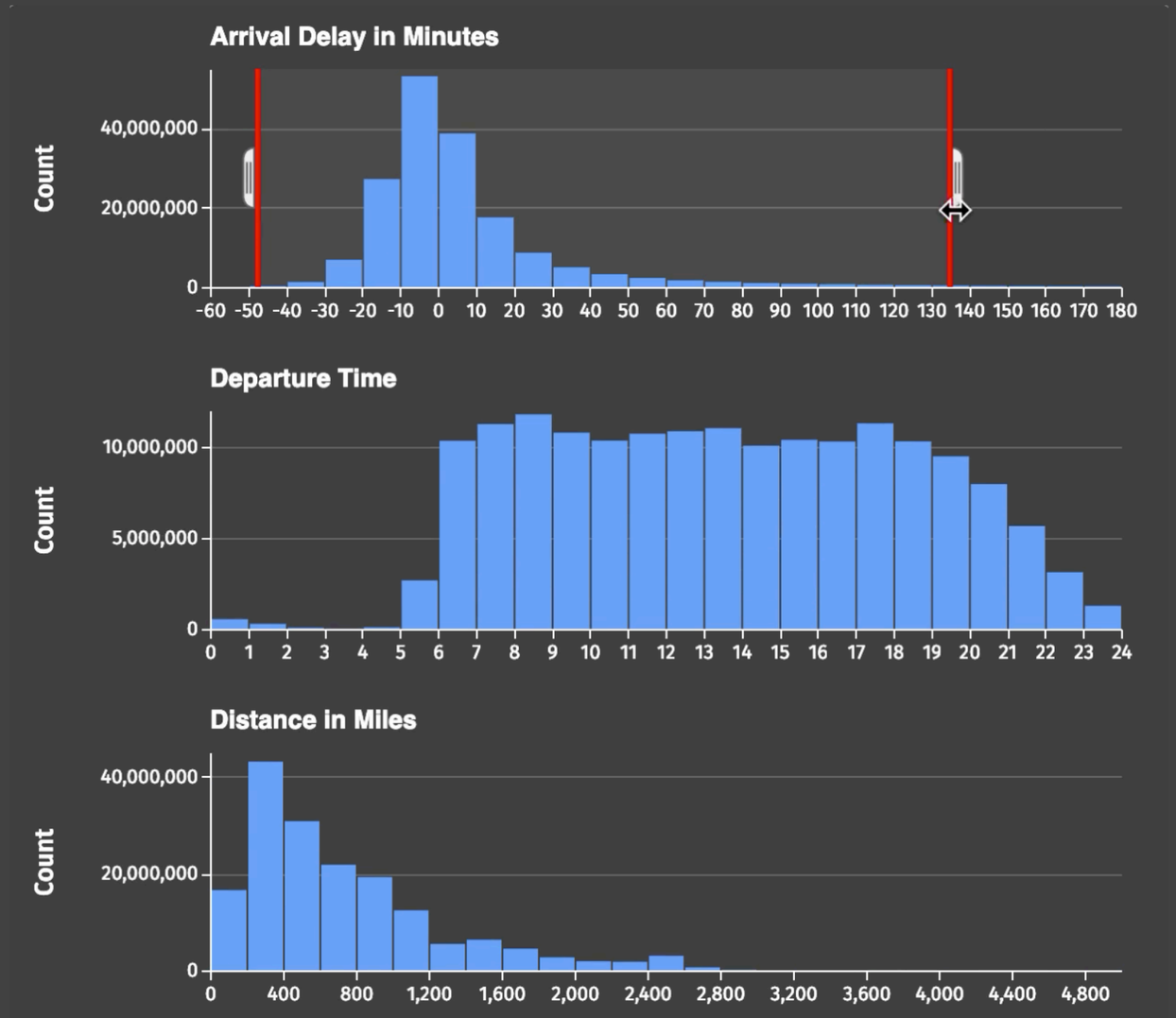
Re-compute if the user switches the view.



brushes in the precomputed view

serves requests from a data cube

Data Cube. Gray et al. *1997.*



brushes in the precomputed view

serves requests from a data cube

Data Cube. Gray et al. *1997.*

interacts with a new view

query for new data cubes

Constant data & time.
Client only.

brushes in the precomputed view

serves requests from a data cube

Data Cube. Gray et al. *1997.*

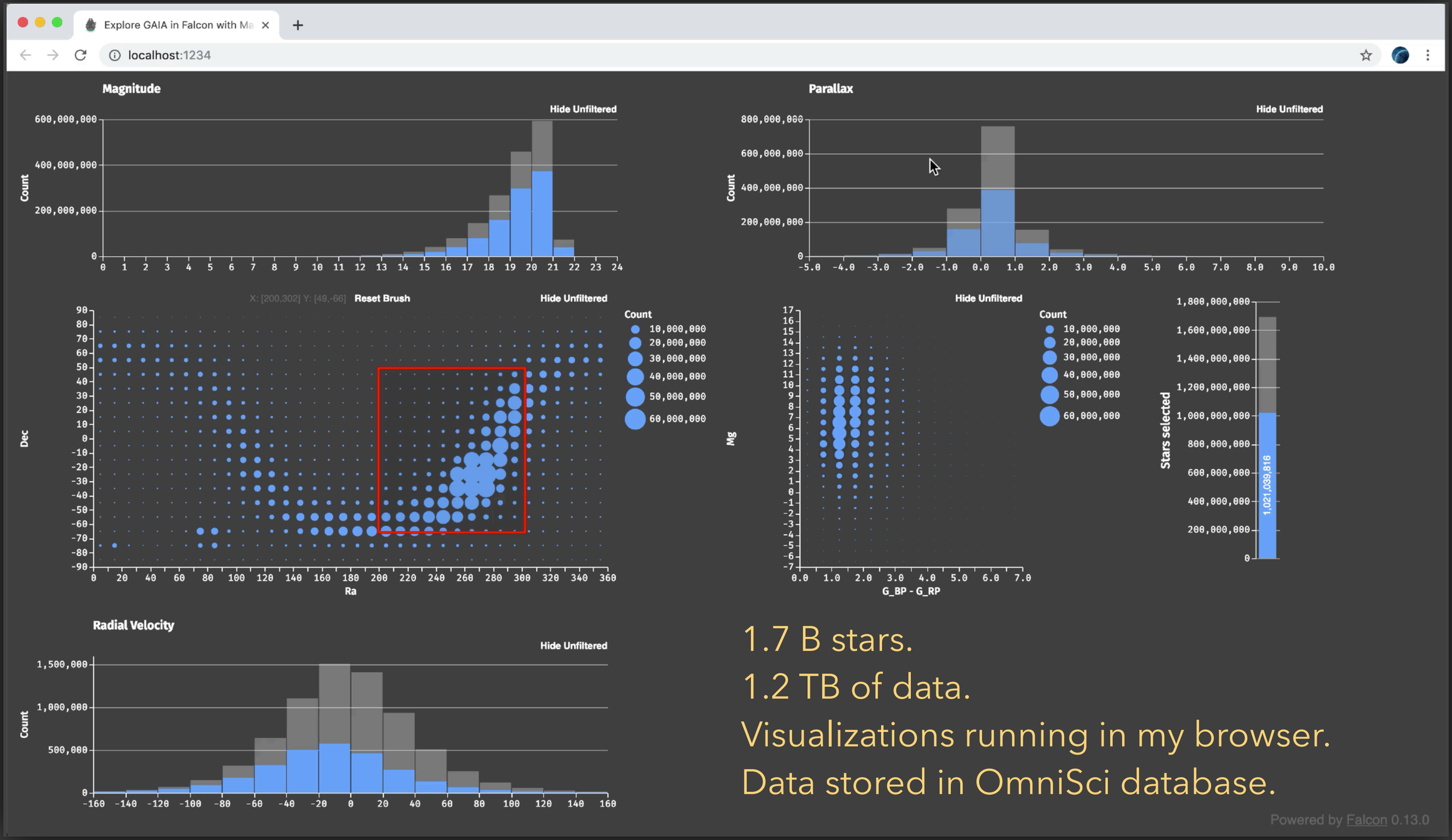
💡 Aggregation decouples interactions from queries over the raw data.

Requires one pass
over the data.

interacts with a new view

query for new data cubes

💡 View switches are **rare** and users are **not as latency sensitive** with them.


Explore GAIA in Falcon with Ma
localhost:1234
Magnitude
Hide Unfiltered
Count
600,000,000
400,000,000
200,000,000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Parallax
Hide Unfiltered
Count
800,000,000
600,000,000
400,000,000
200,000,000
0
-5.0 -4.0 -3.0 -2.0 -1.0 0.0 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
X: [200,302] Y: [49,-66]
Reset Brush
Hide Unfiltered
Count
10,000,000
20,000,000
30,000,000
40,000,000
50,000,000
60,000,000
Dec
90 80 70 60 50 40 30 20 10 0 -10 -20 -30 -40 -50 -60 -70 -80 -90
0 20 40 60 80 100 120 140 160 180 200 220 240 260 280 300 320 340 360
Ra
Hide Unfiltered
Count
10,000,000
20,000,000
30,000,000
40,000,000
50,000,000
60,000,000
Mg
17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -4 -5 -6 -7
0.0 1.0 2.0 3.0 4.0 5.0 6.0 7.0
G_BP - G_RP
Stars selected
1,800,000,000
1,600,000,000
1,400,000,000
1,200,000,000
1,000,000,000
800,000,000
600,000,000
400,000,000
200,000,000
0
1,021,039,816
Radial Velocity
Hide Unfiltered
Count
1,500,000
1,000,000
500,000
0
-160 -140 -120 -100 -80 -60 -40 -20 0 20 40 60 80 100 120 140 160
1.7 B stars.
1.2 TB of data.
Visualizations running in my browser.
Data stored in OmniSci database.
Powered by Falcon 0.13.0

"With Falcon it feels like I'm really interacting with my data."

Data Platform Engineer at Stitch Fix

In Conclusion…

Two Challenges:

1. Effective **visual encoding**
2. Real-time **interaction**

Perceptual and interactive scalability should be limited by the **chosen resolution** of the visualized data, not the number of records.

Bin > Aggregate (**> Smooth**) **> Plot**

1. **Bin** Divide data domain into discrete “buckets”
2. **Aggregate** Count, Sum, Average, Min, Max, ...
3. **Smooth** *Optional*: smooth aggregates [Wickham ’13]
4. **Plot** Visualize the aggregate values

Interactive Scalability Strategies

1. **Query Database**
2. **Client-Side Indexing / Data Cubes**
3. **Prefetching**
4. **Approximation**

These strategies are **not** mutually exclusive!
Systems can apply them in tandem.