UpSet 2.0 and Provenance Library

Kiran Gadhave

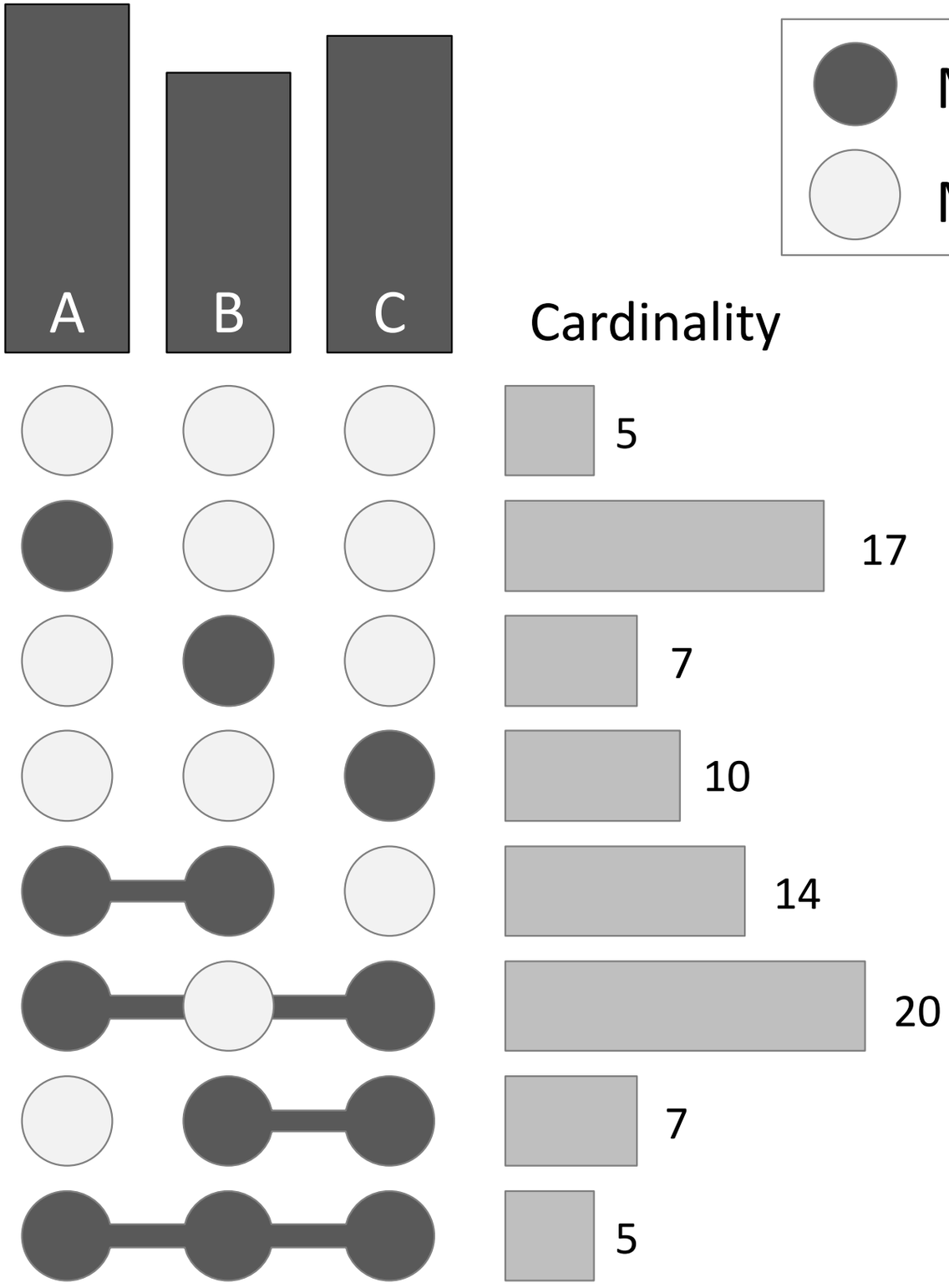
UpSet: Visualization of Intersecting Sets

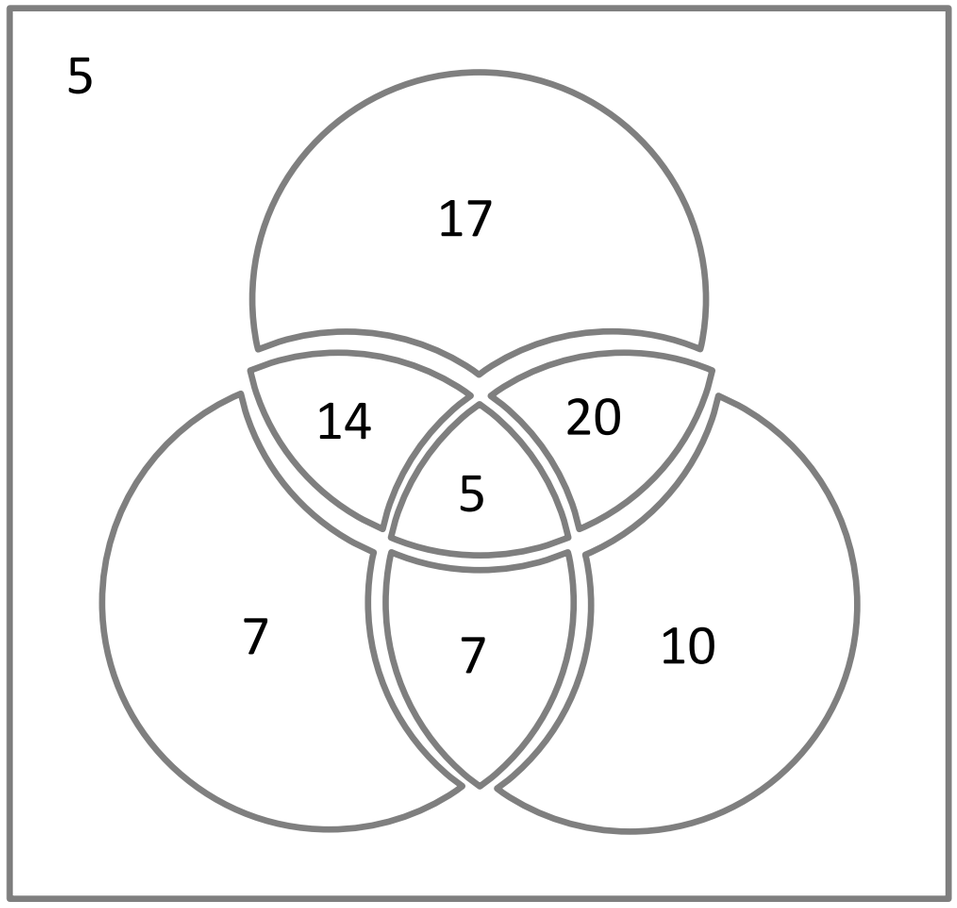
- Originally published in InfoVis ’14 by Alexander Lex, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, Hanspeter Pfister

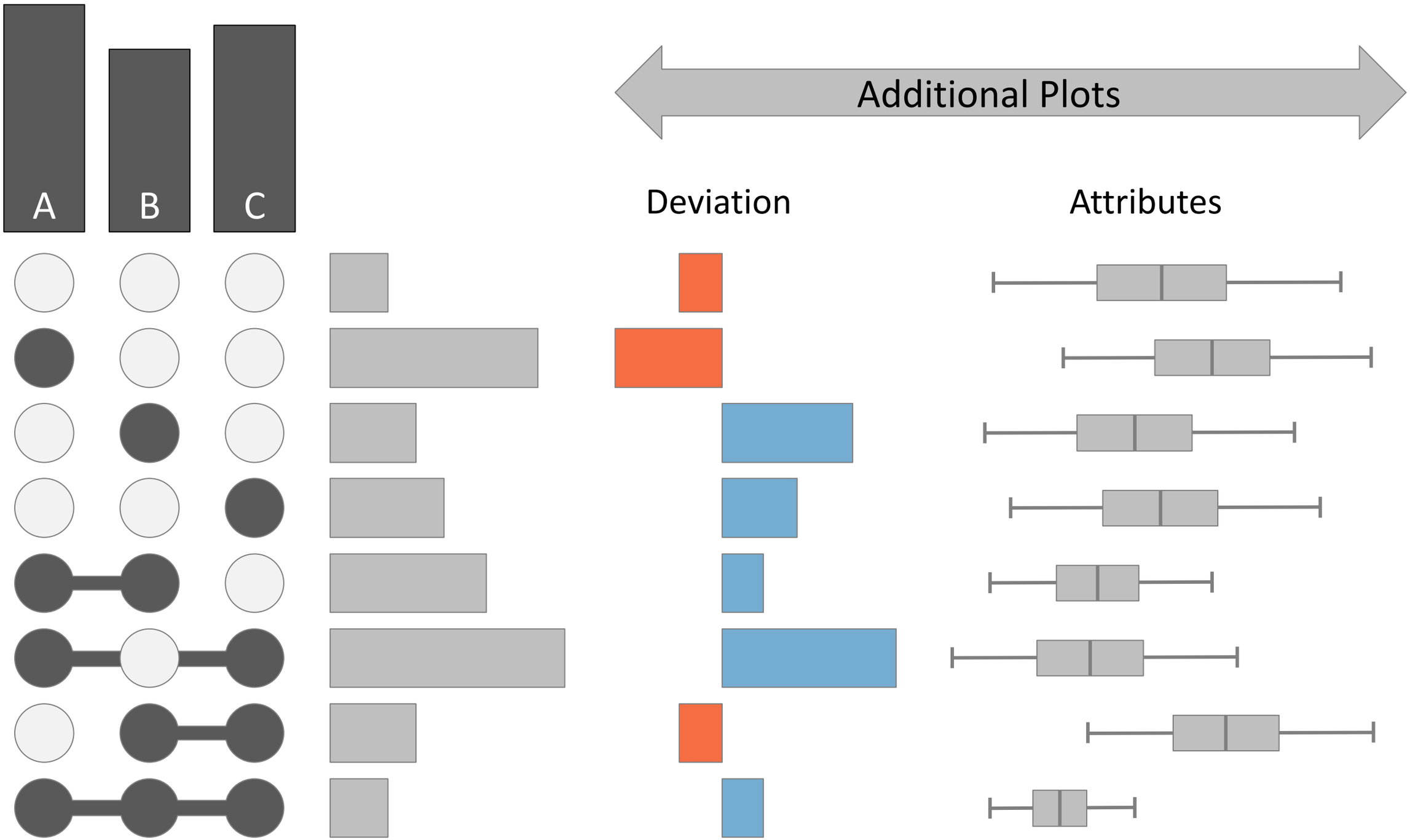
- R version of this technique, UpSetR, was published in 2017 and is quite popular in biomedical domain.

- Introduces a new technique to visualize set intersections combining a set view and element view.

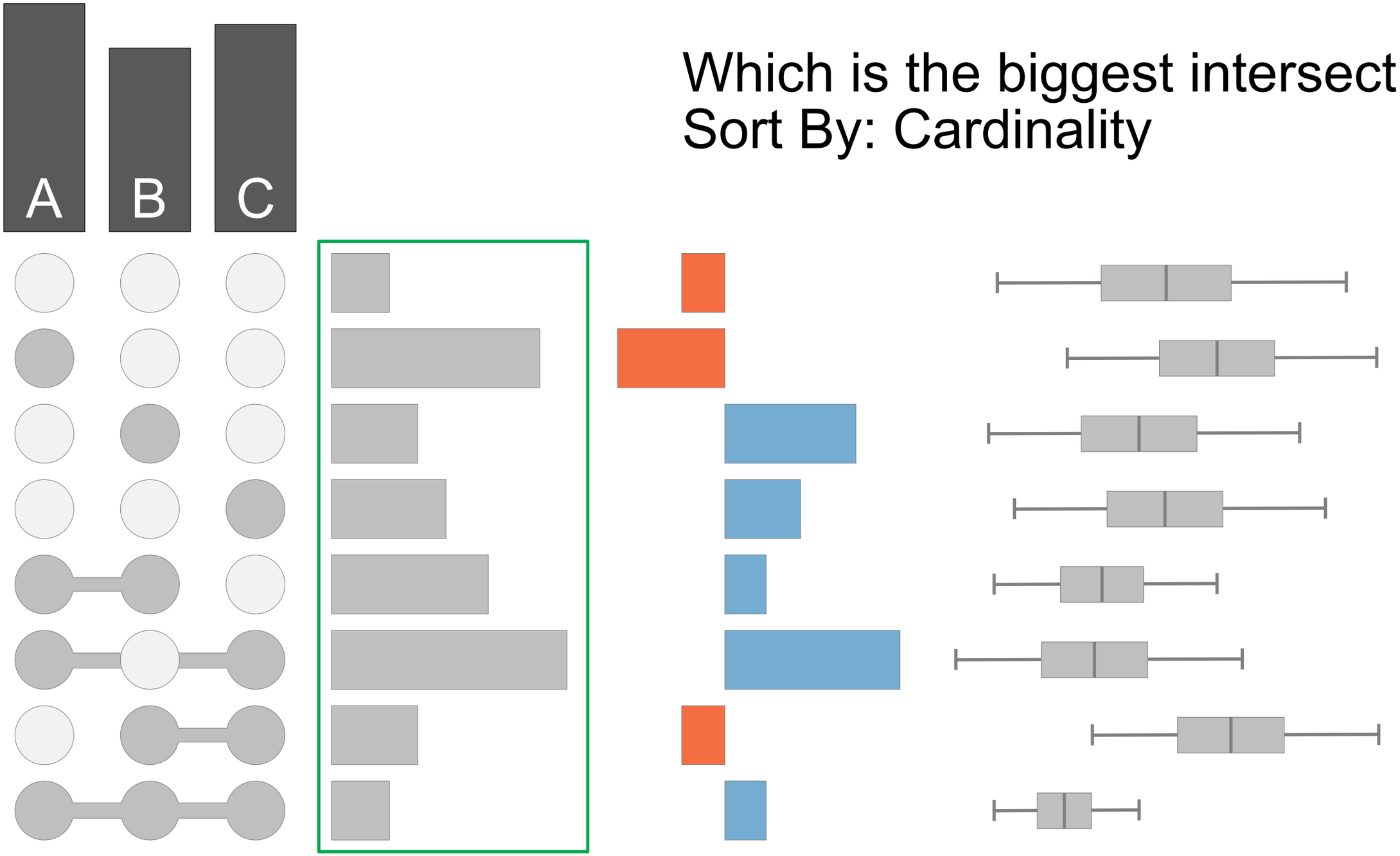
The Technique


A
B
C
Must
Must Not
Cardinality
5
17
7
10
14
20
7
5

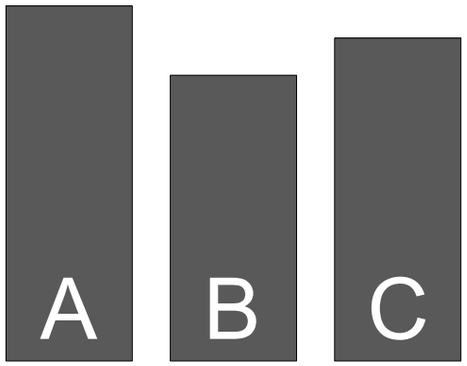

5
17
14
20
5
7
7
10


A
B
C
Additional Plots
Deviation
Attributes

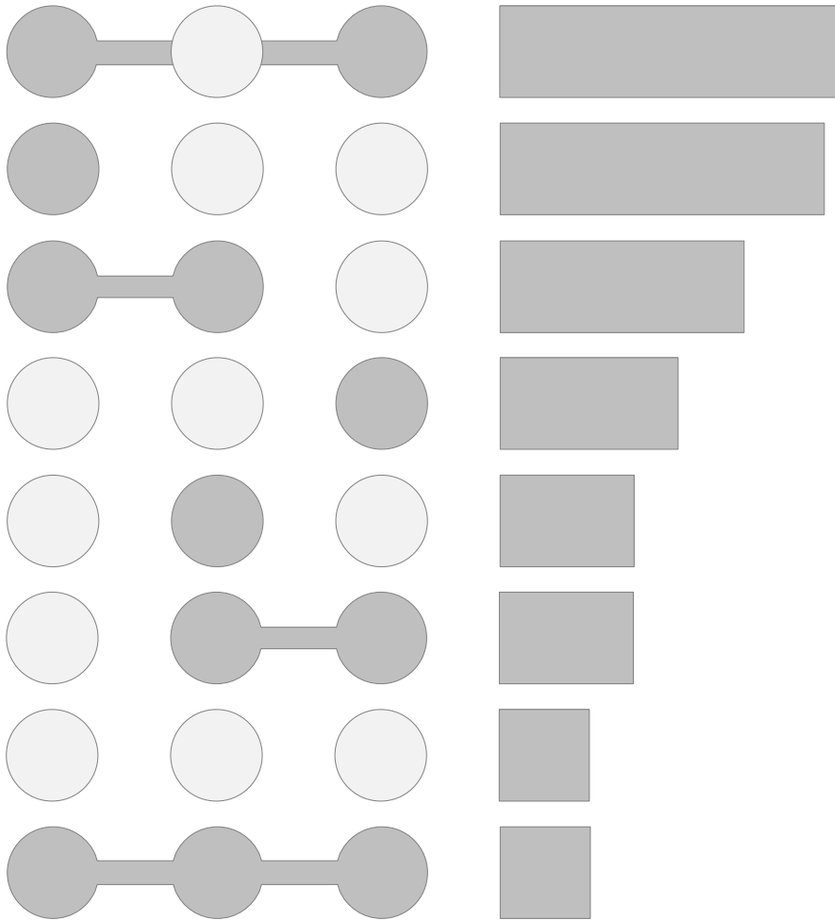
Sorting

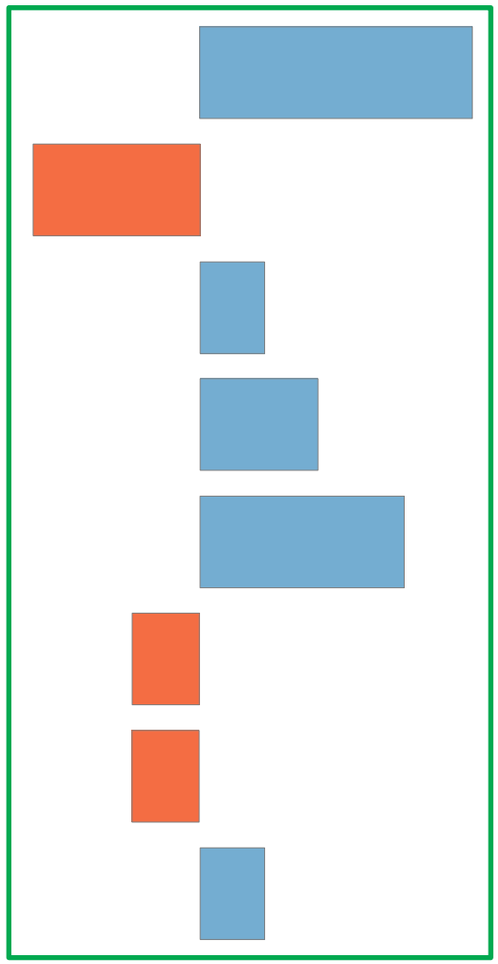


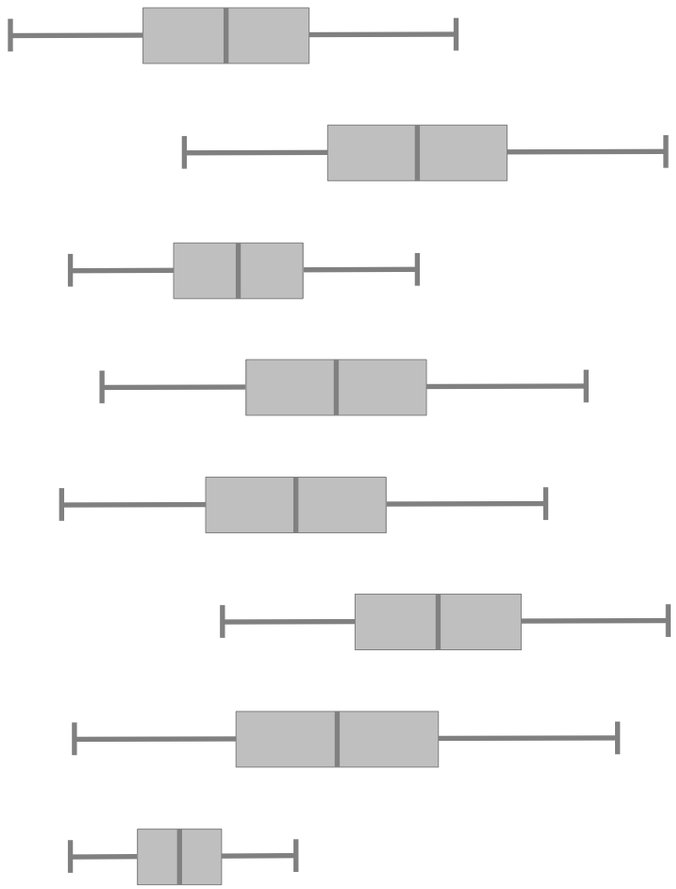
Which is the biggest intersection?
Sort By: Cardinality

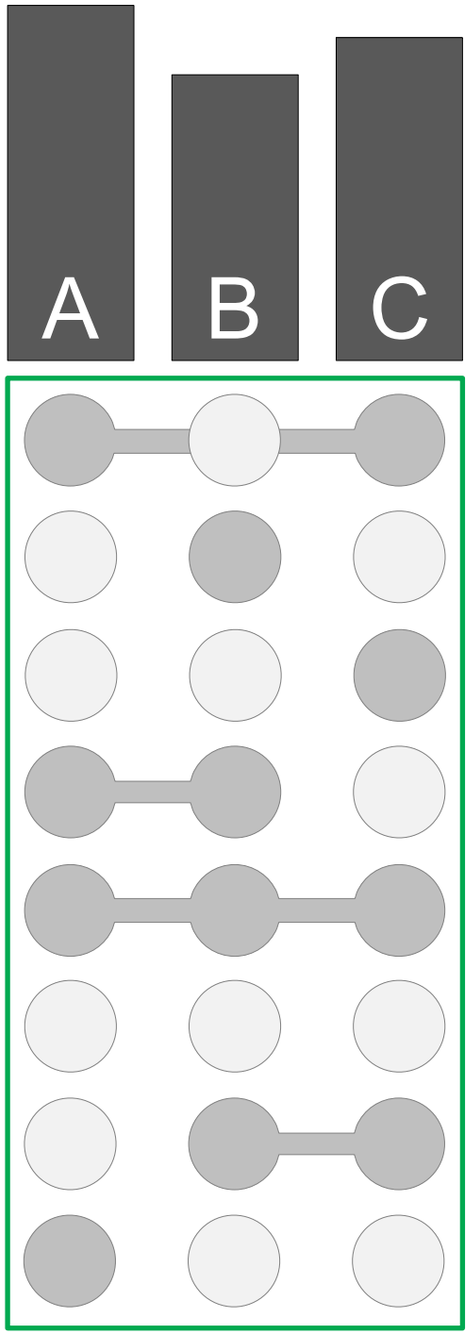


Which is the most ‘surprising’ intersection?
Sort By: Deviation



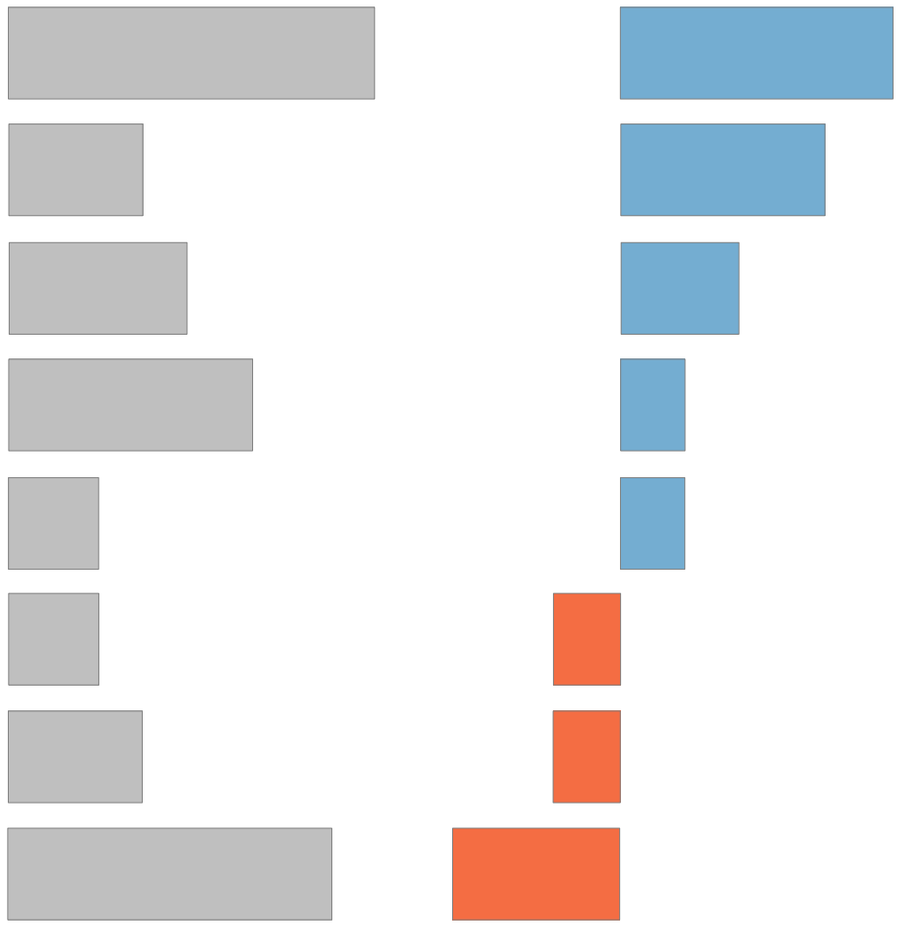


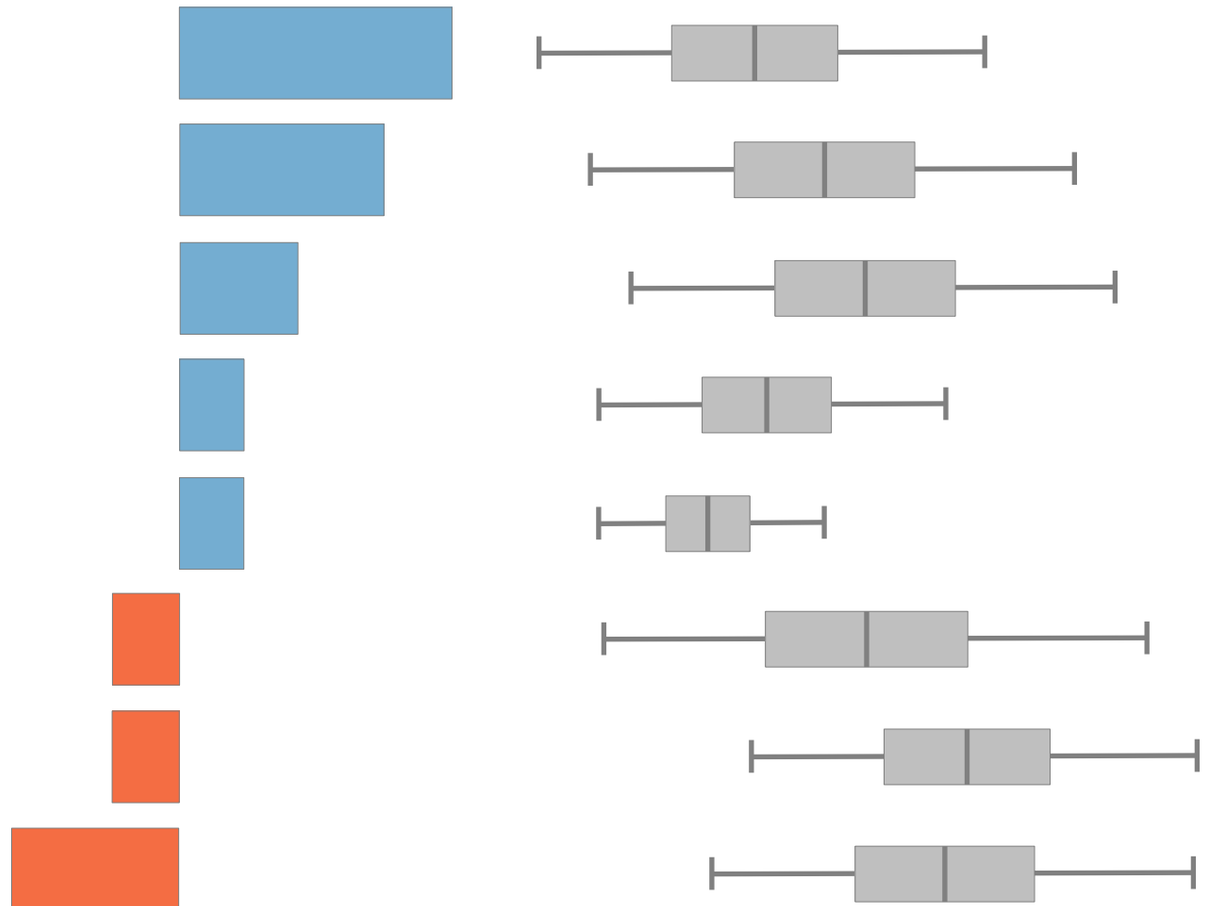




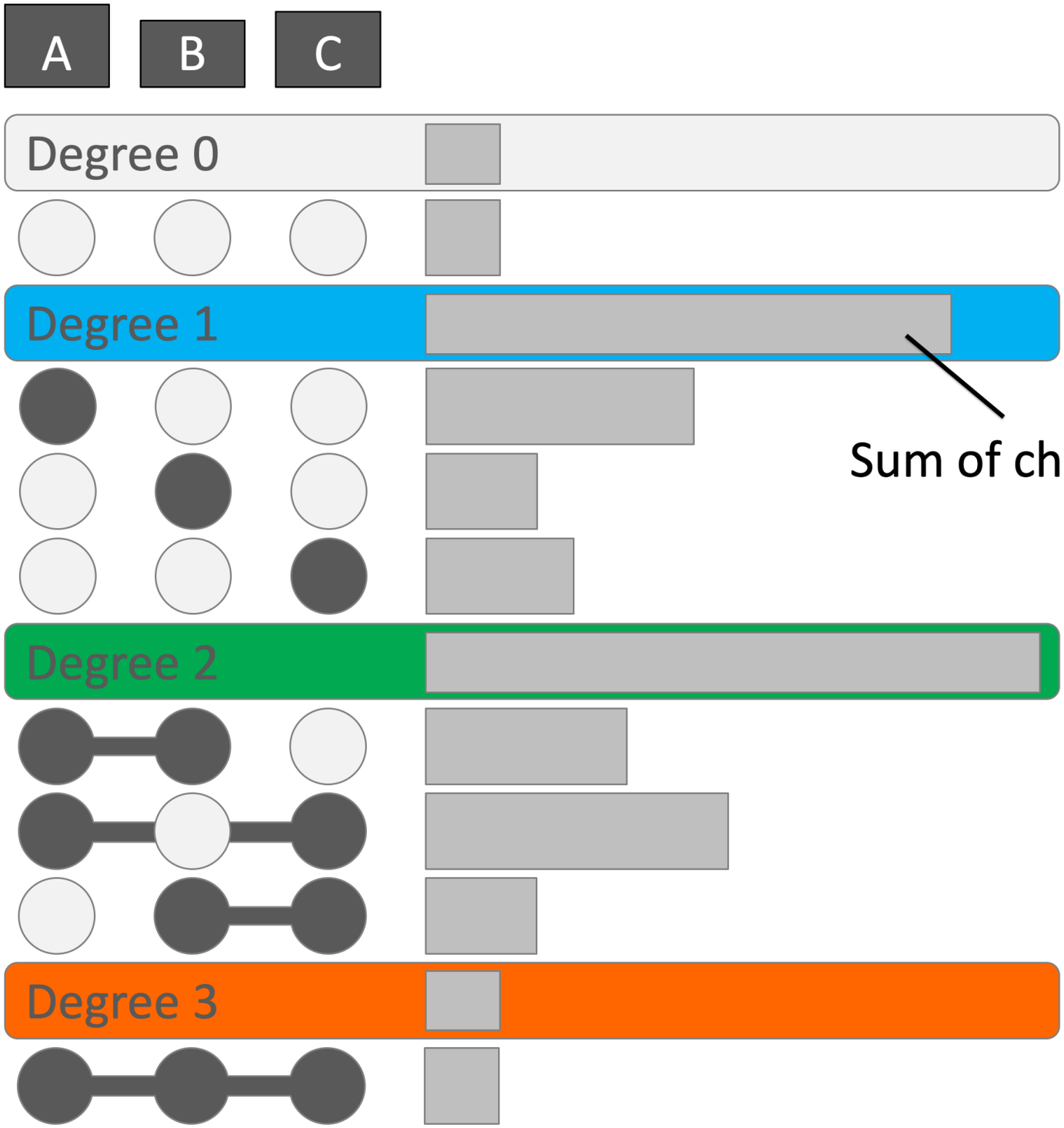
What are the properties of the intersections involving ‘A’?

Sort By: Set



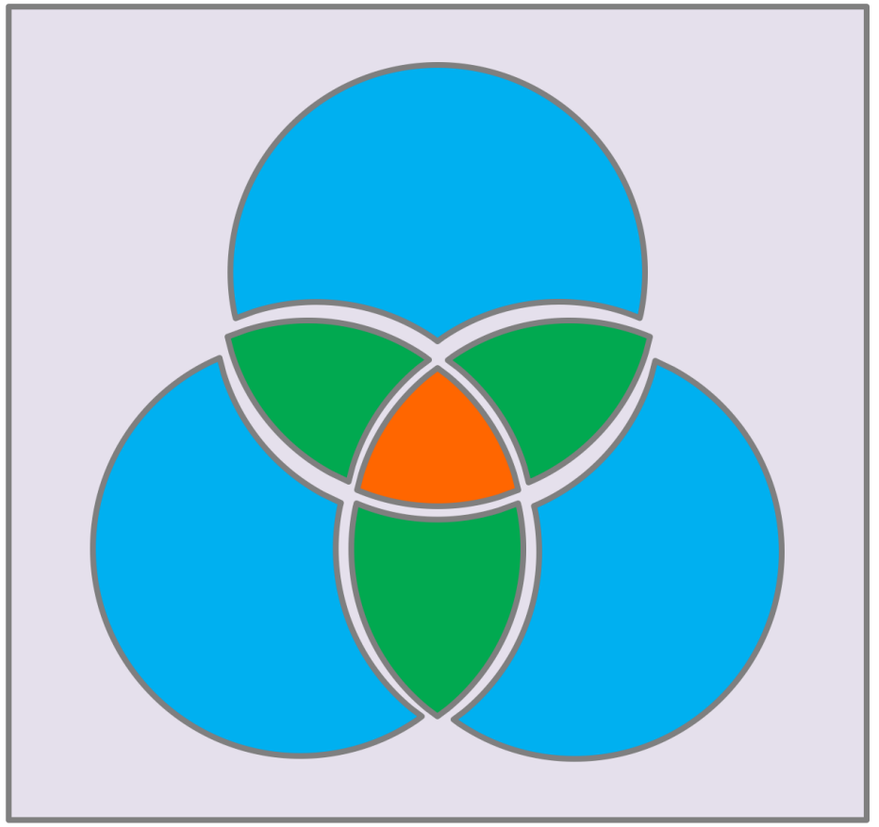


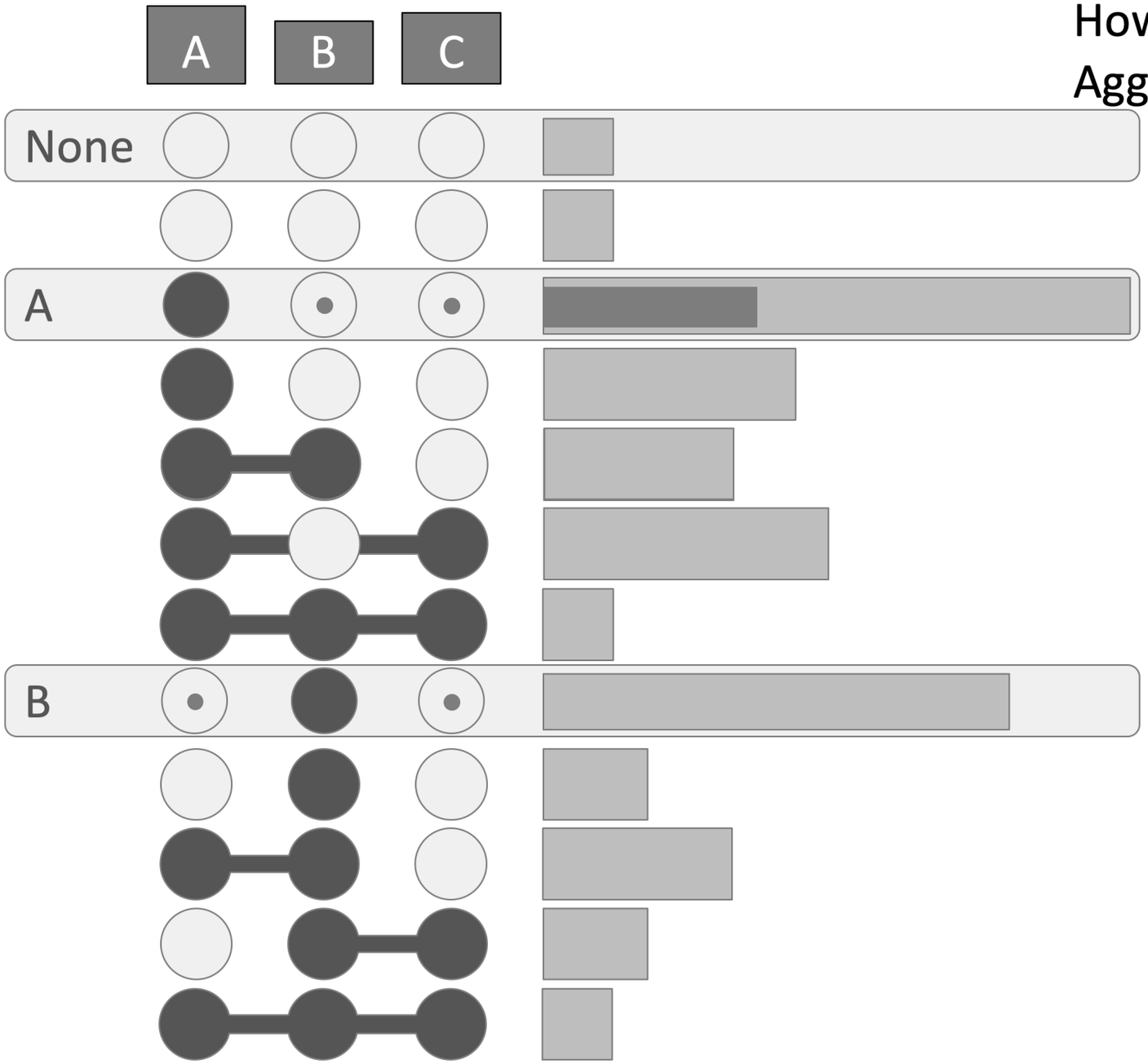
Aggregation



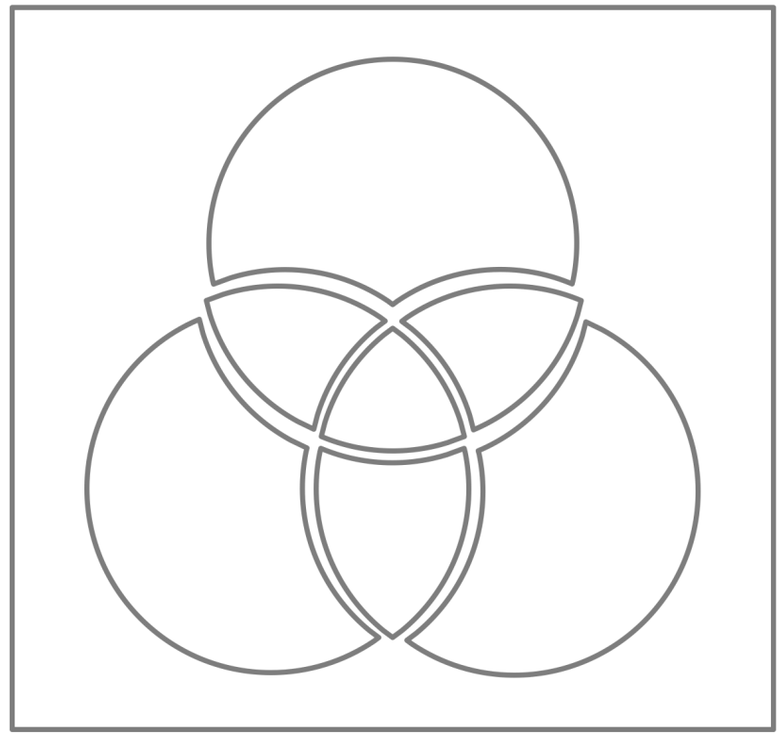
Are many items shared between two sets?
Aggregate By: Degree

Sum of children





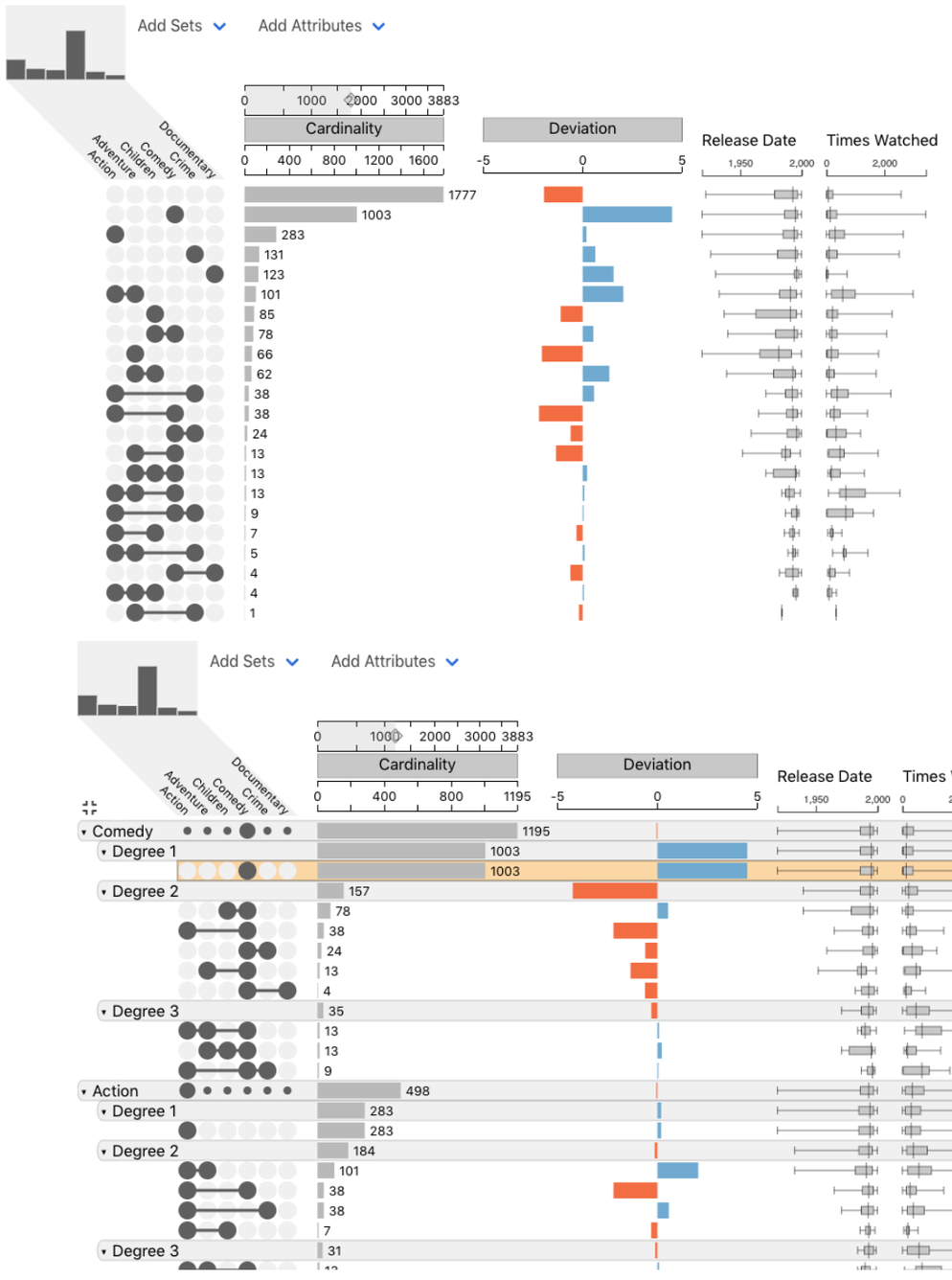
How are the elements of ‘B’ distributed?
Aggregate By: Set



Summary

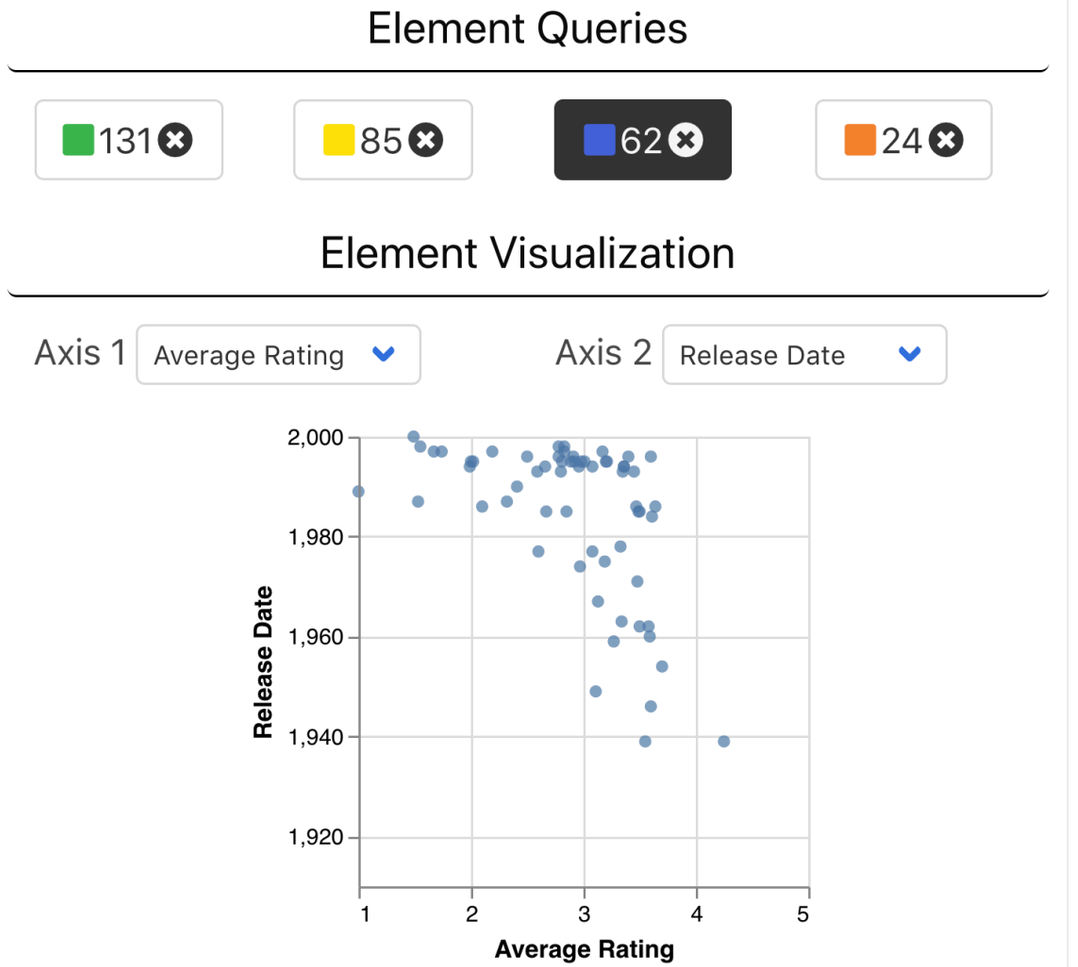
Matrix View

- Visualizes set intersections in a matrix layout.
- Aggregate and visualize the properties based on set intersections, groups and queries.
- Can create queries for element view by clicking on rows of matrix view.
- Elements can be sorted by degree, cardinality or deviation.



Element View

- Shows a list of queries with active query highlighted.
- Visualizes the elements in selected queries in a scatter plot.
- Individual elements are shown in a table, and can be downloaded.



Query Results

Name	Release Date	Average Rating	Times Watched	Set Count
Jumanji (1995)	1995	3.2	701	3
Tom and Huck (1995)	1995	3.01	68	2
Kids of the Round Table (1995)	1995	2	9	3

UpSet - Visualizing Intersecting Sets | Set Overlaps | Choose Dataset: Movies Genres (17 sets, 4 attributes) | Load Data | Embed | About UpSet

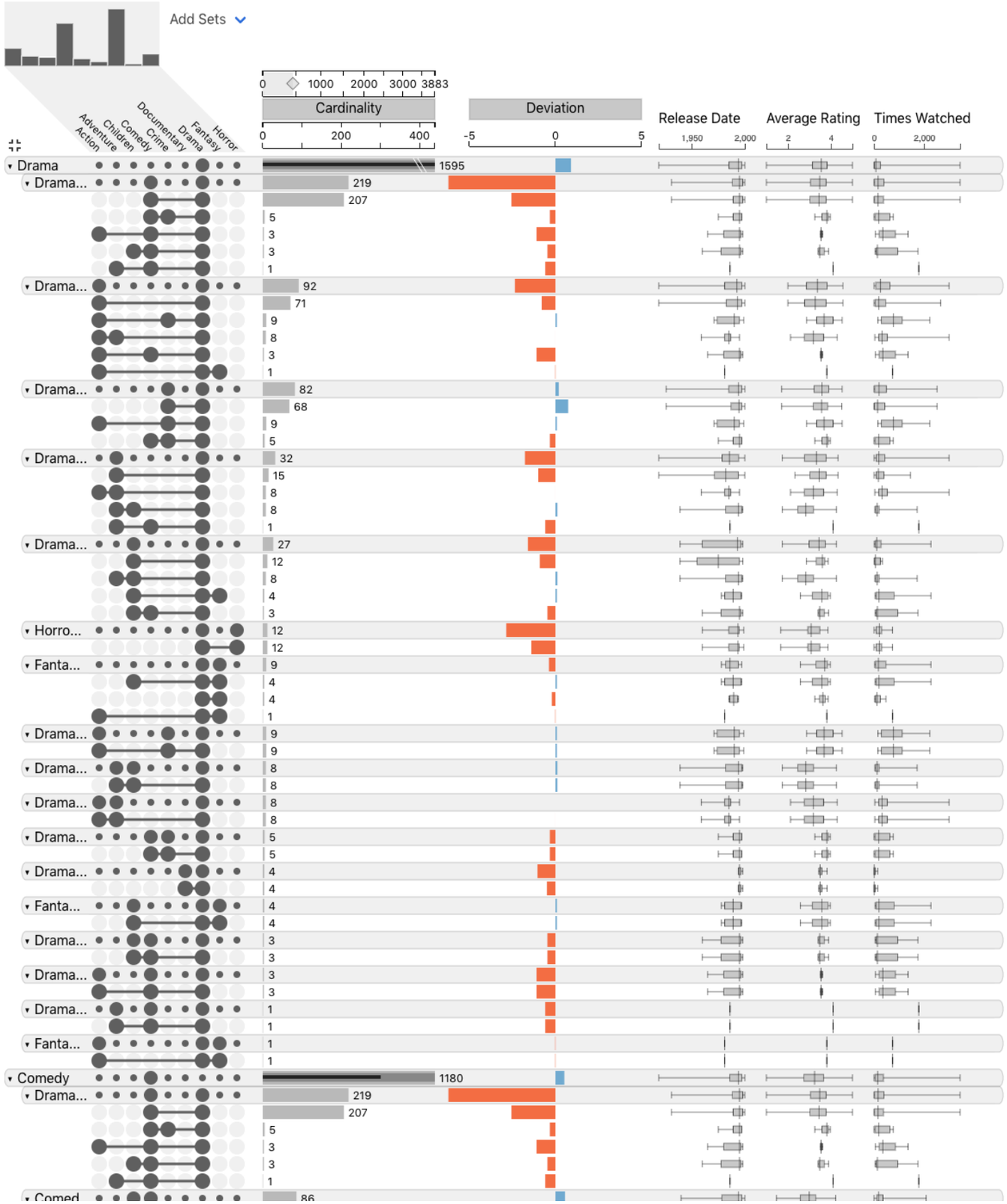
First Aggregate By:
- Degree
- Sets
- Deviation
- Overlaps
- None

Then Aggregate By:

Sort By:

Data

Dataset Information
Name: Movies Genres
 # Sets: 17
 # Attributes: 4
Author: grouplens
Description: MovieLens ratings dataset, curated and filtered by Alsallakh.
Source:
http://grouplens.org/datasets/movielens/

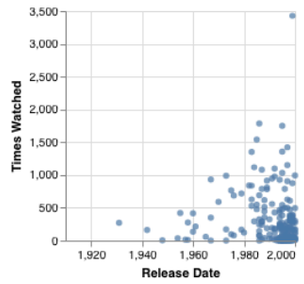


Element Queries

9 | 9 | 1 | 4 | 219

Element Visualization

Axis 1: Release Date | Axis 2: Times Watched



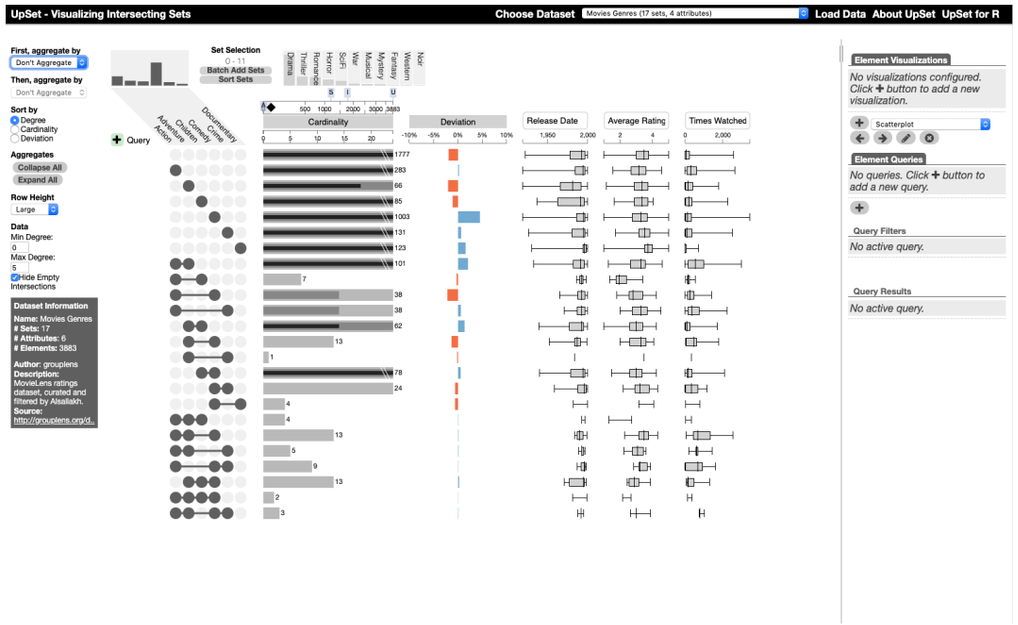
Query Filters

Query Results

Name	Release Date	Average Rating	Times Watched	Set Count
Waiting to Exhale (1995)	1995	2.73	170	2
American President, The (1995)	1995	3.79	1033	3
To Die For (1995)	1995	3.42	544	2
Kicking and Screaming (1995)	1995	3.63	93	2
Big Bully (1996)	1996	1.92	12	2
Last Summer in the Hamptons (1995)	1995	3.1	21	2
Nobody Loves Me (Keiner liebt mich) (1994)	1994	3.58	12	2
Nueba Yol (1995)	1995	1	1	2
Doom Generation, The (1995)	1995	2.46	61	2
Something to Talk About (1995)	1995	2.84	164	3
Unstrung Heroes (1995)	1995	3.61	85	2
Boys on the	1995	3.13	195	2

New in UpSet 2.0

- Entirely re-implemented in **TypeScript**.
- Improved user interface

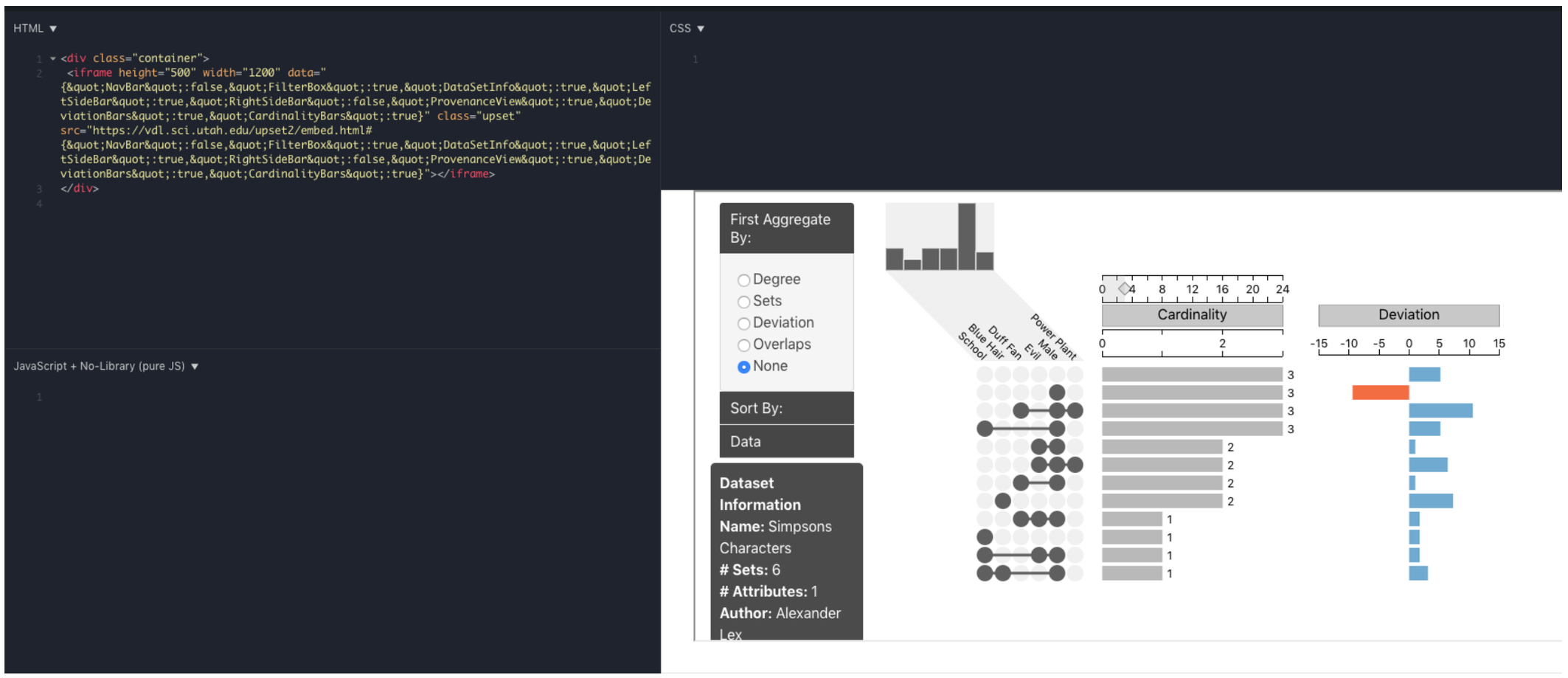


UpSet

UpSet 2.0

New in UpSet 2.0

- Can use UpSet technique as a TS/JS **library**.
- Can be **embedded** in existing webpage as an iFrame. Below is an example of embedding as Iframe on JSFiddle.



Requires no JS or CSS configuration when embedding

New in UpSet 2.0

- Can **download** the subset of data which matches the queries.

Query Results

Name	Release Date	Average Rating	Times Watched	Set Count
Tigrero: A Film That Was Never Made (1994)	1994	3.5	4	2
Looking for Richard (1996)	1996	3.82	146	2
JLG/JLG - autoportrait de d?embre (1994)	1994	3.5	4	2
Endurance (1998)	1998	3.43	14	2

- Supports Undo/Redo using **Provenance tracking**.
- Demo

Provenance

Provenance…

The execution history of processes which were utilized to compute a final piece of data.

…in Visualization?

Record of user interactions with the visualization tool which lead to a particular state.

Applications

- Basic use case would be a simple non-linear undo graph. UpSet2.0 uses a linear version of this for simplicity.

- Provenance graph can be saved and shared as JSON object. This allows sharing of visual analytics process.

- Retrace and extend original analysis.

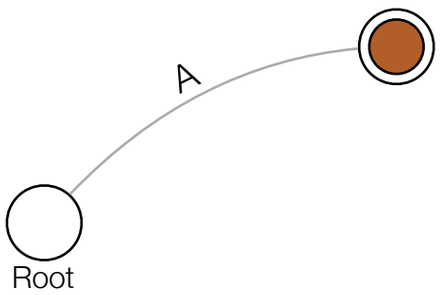
- Interactive presentation of analysis using provenance data.

  - See: From Visual Exploration to Storytelling and Back Again which implements StratomeX and Gapminder as Vistories.

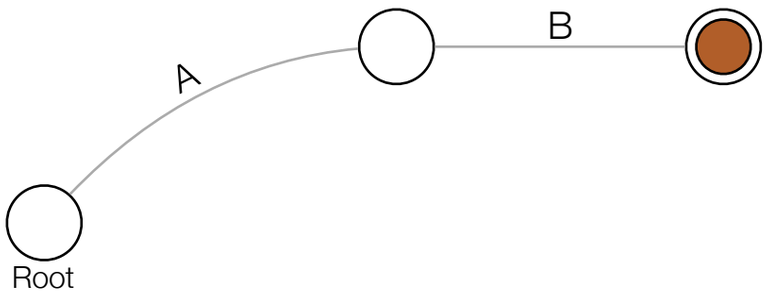
- Analysis of visual analytics process in terms of interactions for automation or optimization.

Root is the default node of the graph.

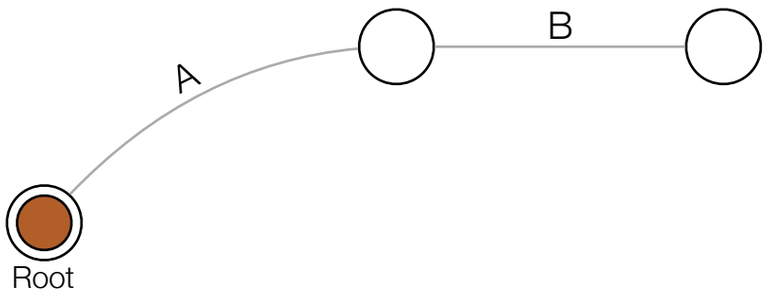
When user performs action A, a new node is added.



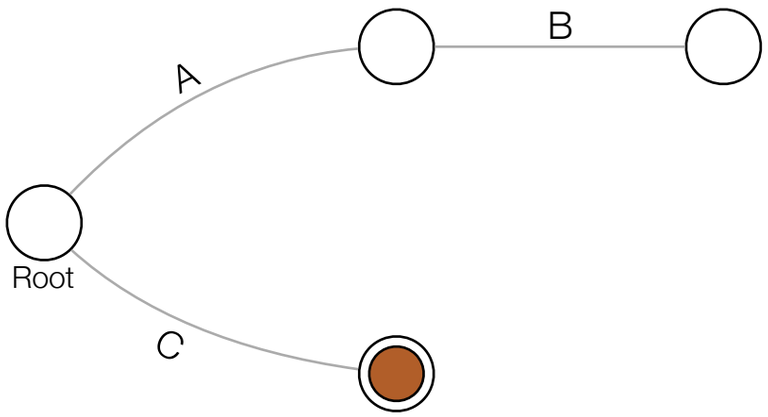
User performs next action B, and another node is added.
Orange circle shows the current node.



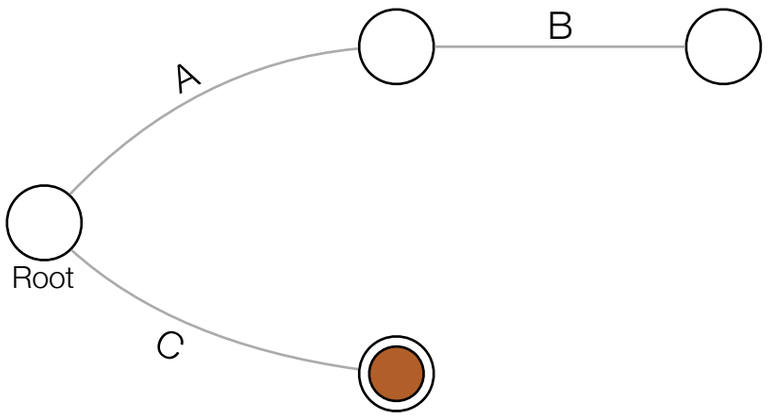
This provenance can be used to go back to any previous state.
When we go back to Root, the actions A and B are undone.



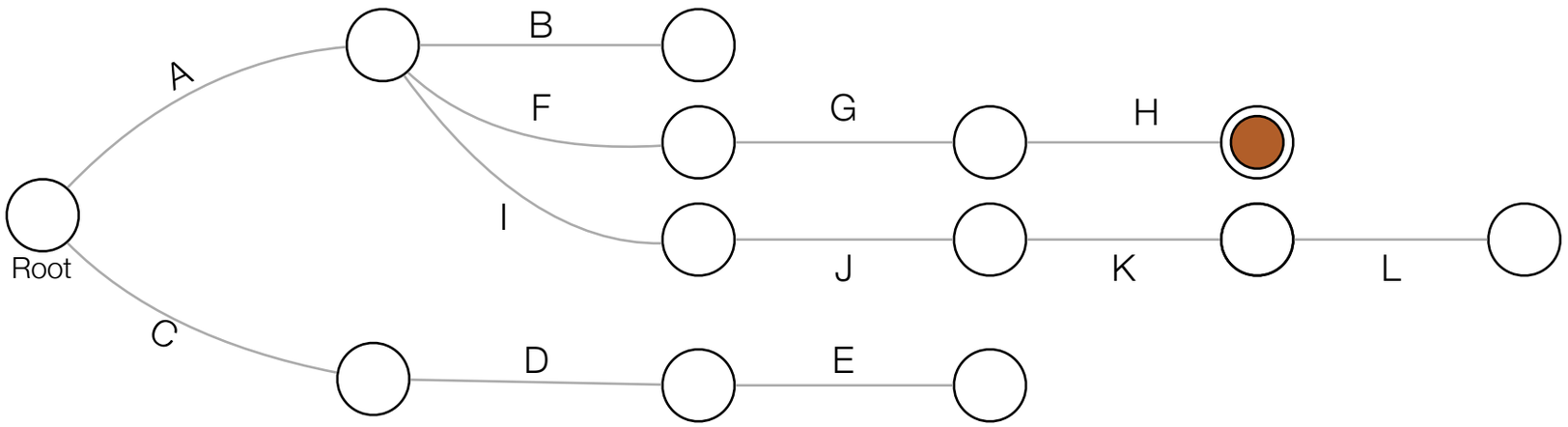
When user performs another action say C, a new branch is started to track from this node.



Branching can take place from any node which already has children.



Branching can take place from any node which already has children.



Provenance Tracking Library

- Github: https://github.com/visdesignlab/provenance-lib-core
- NPM: *npm i provenance-lib-core*
- JavaScript/TypeScript library
- Provides data structures and functions to track interactions in web based tools.
- Records user actions in a graph.

Library

- The provenance graph and tracking is implemented using redux to maintain immutability between states.

- It’s easier to implement provenance tracking in applications which redux to manage states.

- Each action which is trackable must define two parts: DO and UNDO.

- DO part of action is applied when moving away from root and UNDO is applied when moving towards root.

Using the library

- Call Provenance function with redux store of your application as parameter.

- It returns a provenance object which has following functions:

  - apply: This function takes in action to be applied

  - graph: This returns current instance of provenance graph

  - goToNode: This function is used to go to any node in the graph. The path from current node to this node is calculated automatically and proper do/undo actions are applied.

Future Work

- Current provenance tracking applies actions to get to a state. The nodes of the graph do not store any information about state of visualization.

- Currently we are working on storing state of the visualization in the node.

- Advantage of this would be use the stored state directly, in cases where the actions are long running or non-deterministic (say machine learning algorithms).

- Once we start storing states, other extensions could including hybrid approach to store state as well as actions or combine nodes which reach the same state, thereby reducing size of graph.

Thank You

Questions?