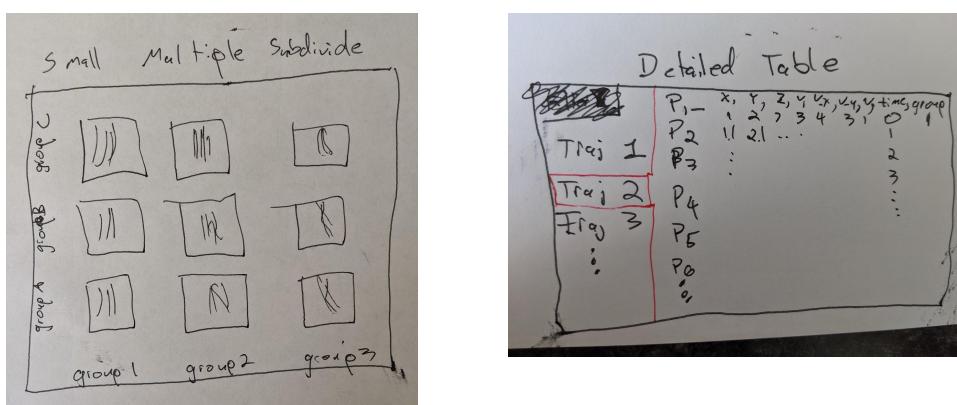
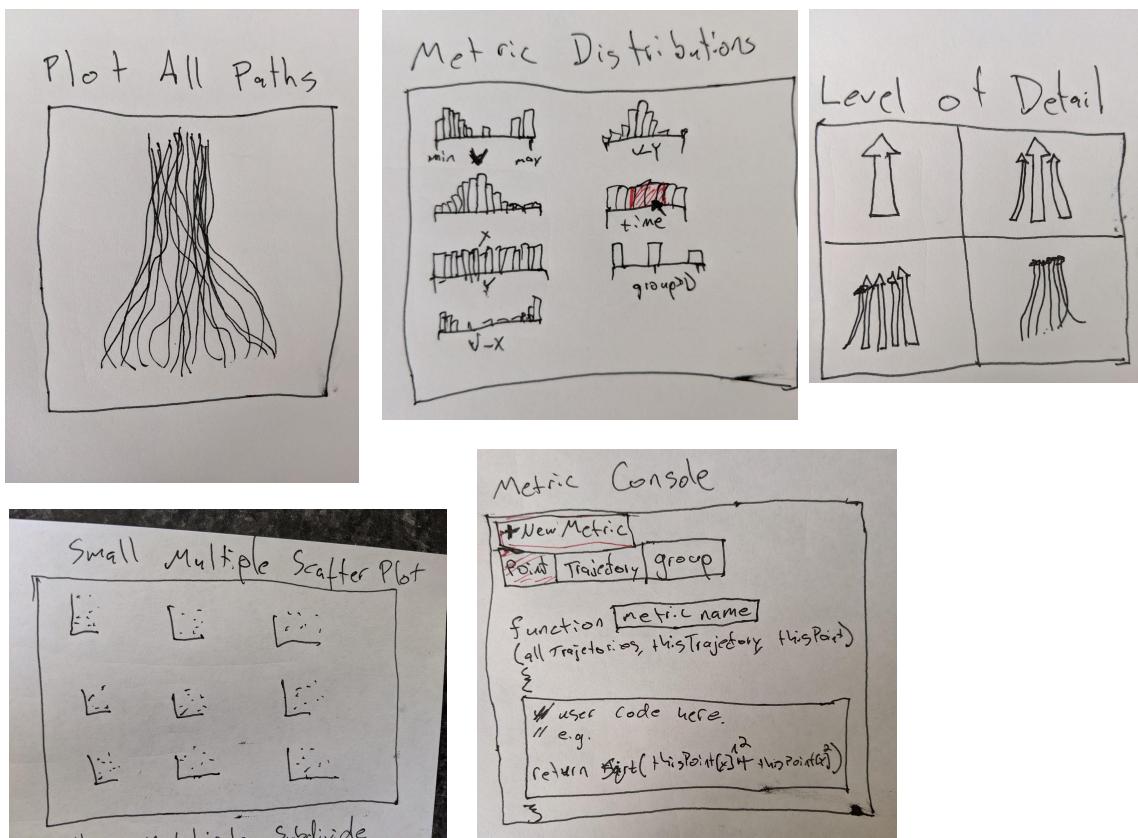


Current Live Prototype: <https://www.devinlange.com/research/MTAT/dist/index.html>

Working:

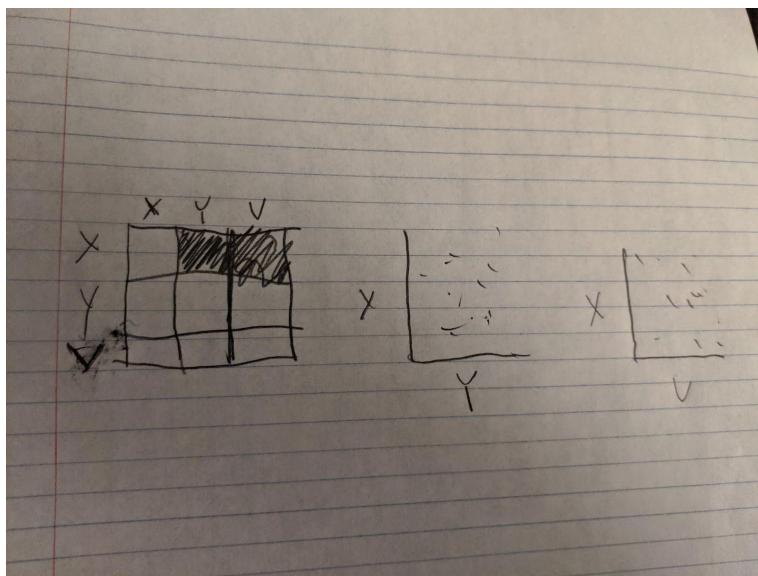
- Data Model
 - Data loading from file
 - Loading from example files
 - Non-interactive Plot of all paths
 - Framework for setting up view and adding different visualizations

My first round of design exploration involved a lot of ideation for different visualizations that could be applied to Multivariate Trajectory data:



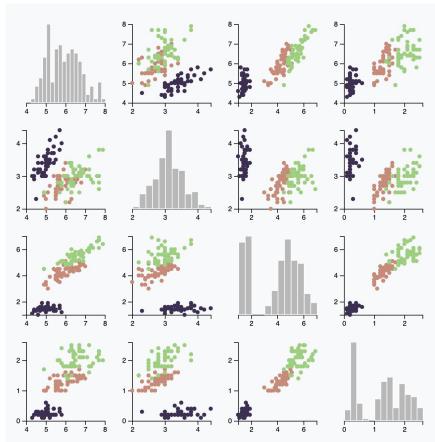
An early discussion about some of these yielded some new insights. First for the level of detail visualization, using a hierarchical clustering technique was another option (other than k-means clustering).

Another discussion was around the scatterplot matrix. I got the feedback that scatterplot matrices that show all combinations are often overkill, and not that useful. Instead it would be better to let the user select the combinations. When thinking through how to let users select which options, I did draw inspiration from the scatterplot matrix for how to let the user select different combinations quickly by giving a matrix they can select and deselect options.

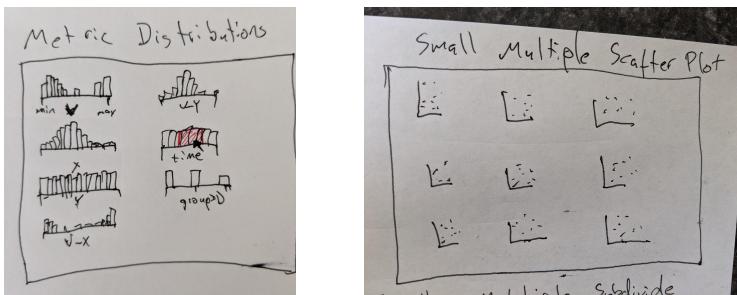


After I had this idea I was exposed to scatterplot matrices with distributions along the diagonal. For example:

From https://www.d3-graph-gallery.com/graph/correlogram_hist.html

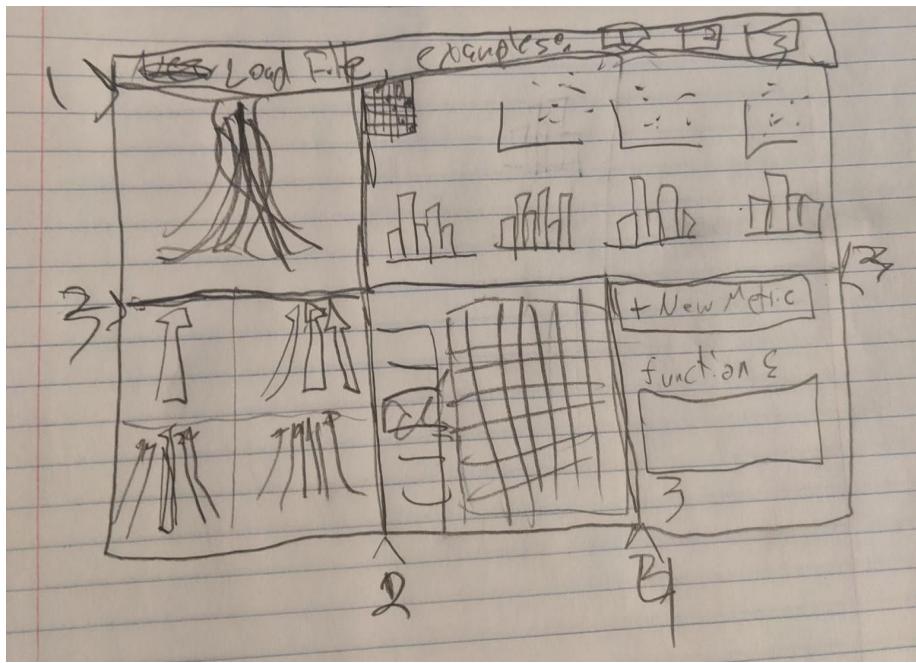


This concept clicked with my widget for selecting scatterplot options, and I decided that I would do something like this, in effect combining these two separate visualization ideas from before into one:



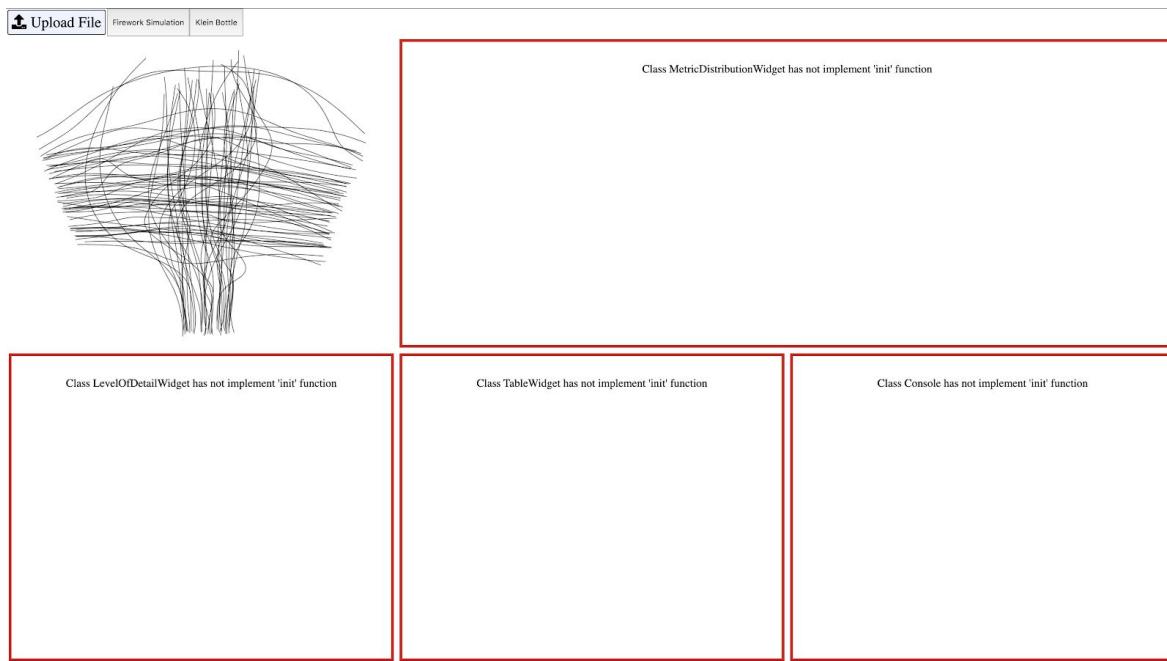
Another possible area of integration of multiple plots is within the tabular view of the data. One nice way of doing this that was pointed out during the visualization feedback session was with scented visualizations. Another separate topic that was discussed at these meetings was the mapping of variables onto the trajectories with color mappings.

As I was writing the code for the layout framework I realized I never combined these sketches into a cohesive layout. I did write the code so that it should be relatively easy to rearrange the layout, but I made an initial layout for the first set of visualization.

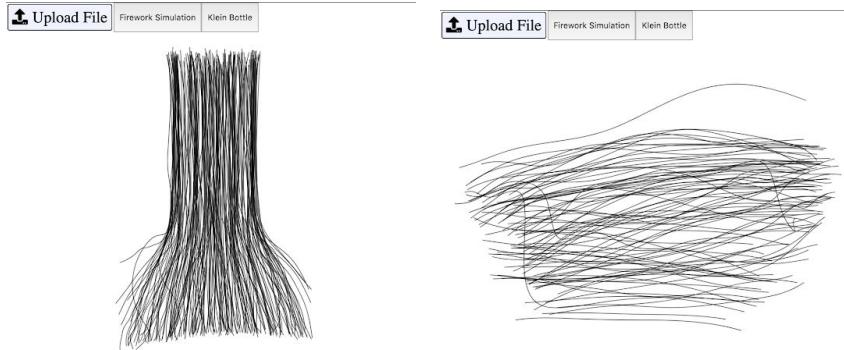


After writing the code for this layout framework as well as at least a stub class for each of the visualizations I can get a more accurately rendered picture of the final idea:

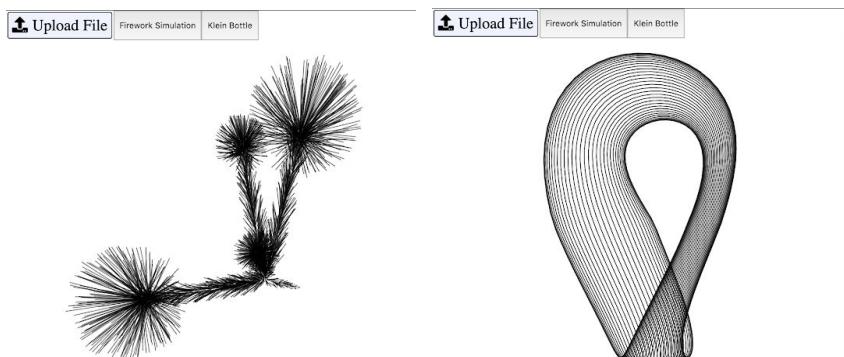
Two groups of people crossing at 90 degrees - with full screen for context:



Other crowd scenarios:



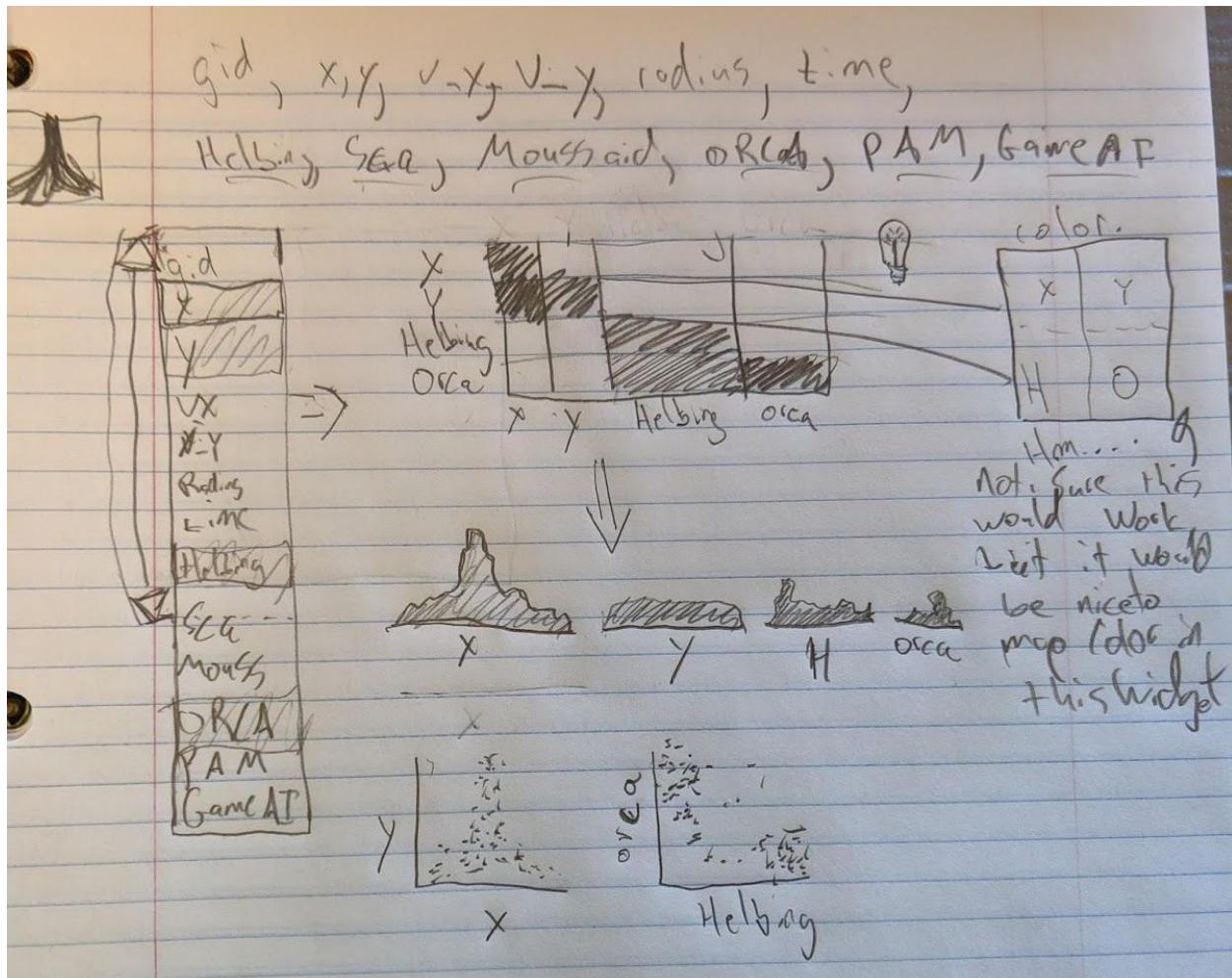
Toy examples:



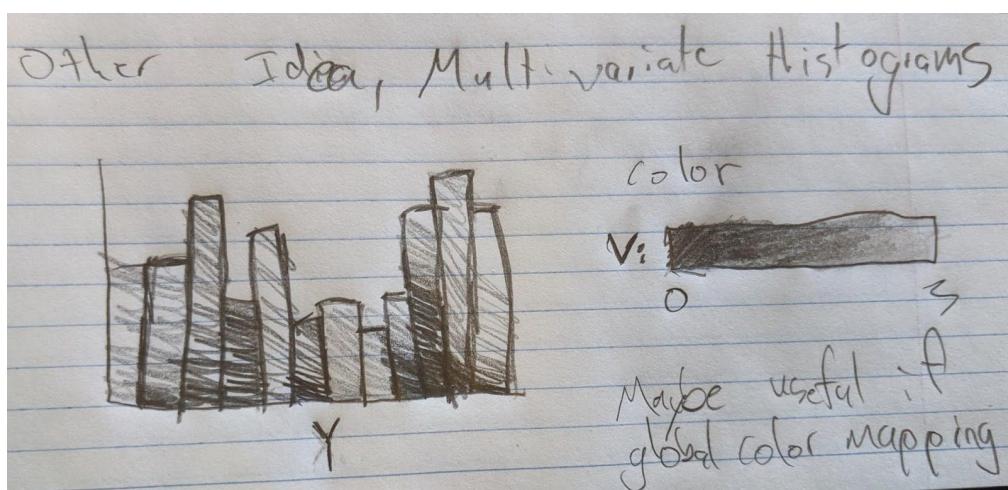
One insight I came on while working through scaling the data so that it keeps its aspect ratio intact is that instead of fitting data inside a box to its fullest extent an argument can be made for shrinking a container to fit the data as tightly as possible.

Todo -> add images illustrating this...

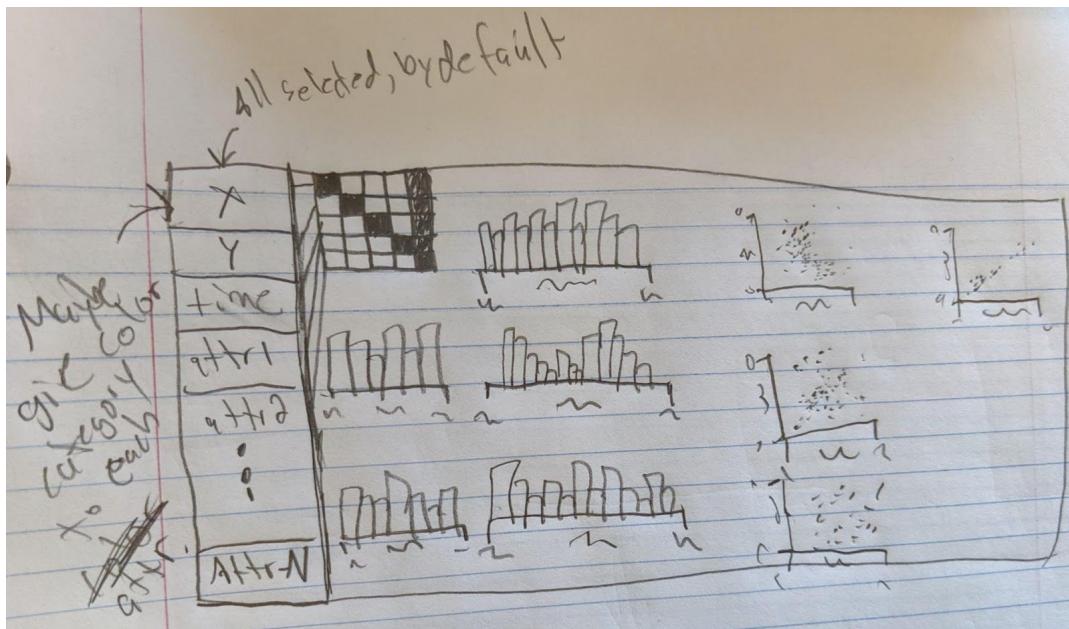
More details on widget for selecting distributions and scatterplots:



Random idea for multivariate histogram. This is probably most useful if there is some kind of global switch for mapping color to a single variable that applies across different widgets.



Here is a litter better sketch of my idea for a widget with histograms and trajectories. Not sure if the minimap in the top left really makes sense or not...



Meeting with Alex:

Topics:

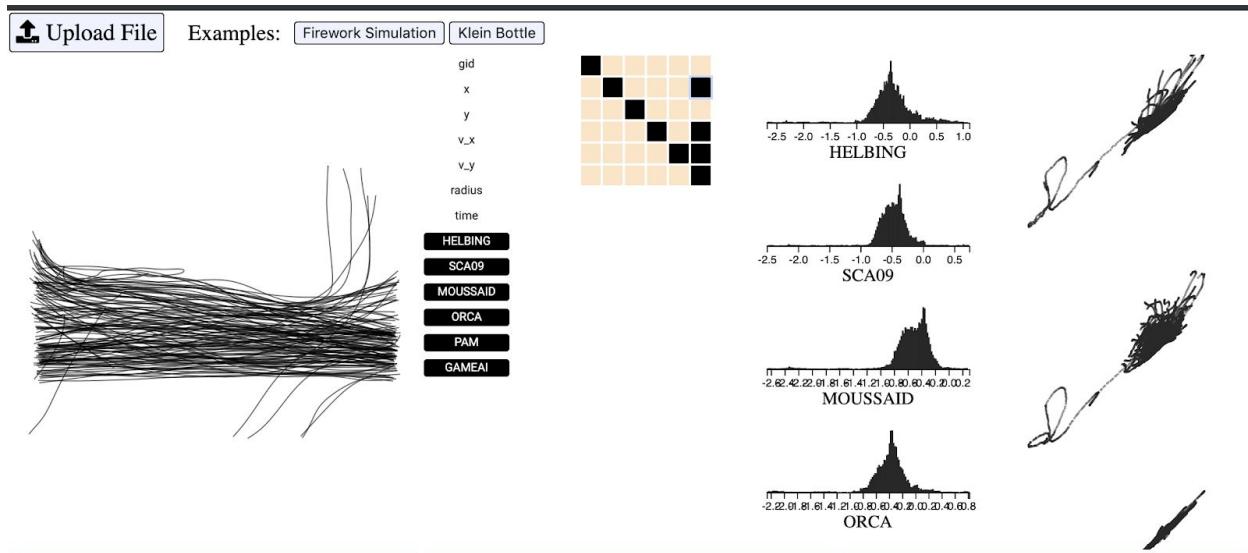
- Logistics
 - Independent Research Sign Up
 - Send a slack message
 - Google calendar access
 - Send reminder in slack
- Viz Project
 - Struggling with what the right consistent metaphor is for:
 - Filtering - completely hiding and showing data
 - Highlighting - make the selection more visible, but don't completely hide other data
 - Selecting - select a single data point and highlight it, and get additional info on it. A data point could be an actual point, or trajectory, or even group of trajectories.

For instance, should the data hide entirely? Should it depend on the viz?
Should it support filtering and selecting?

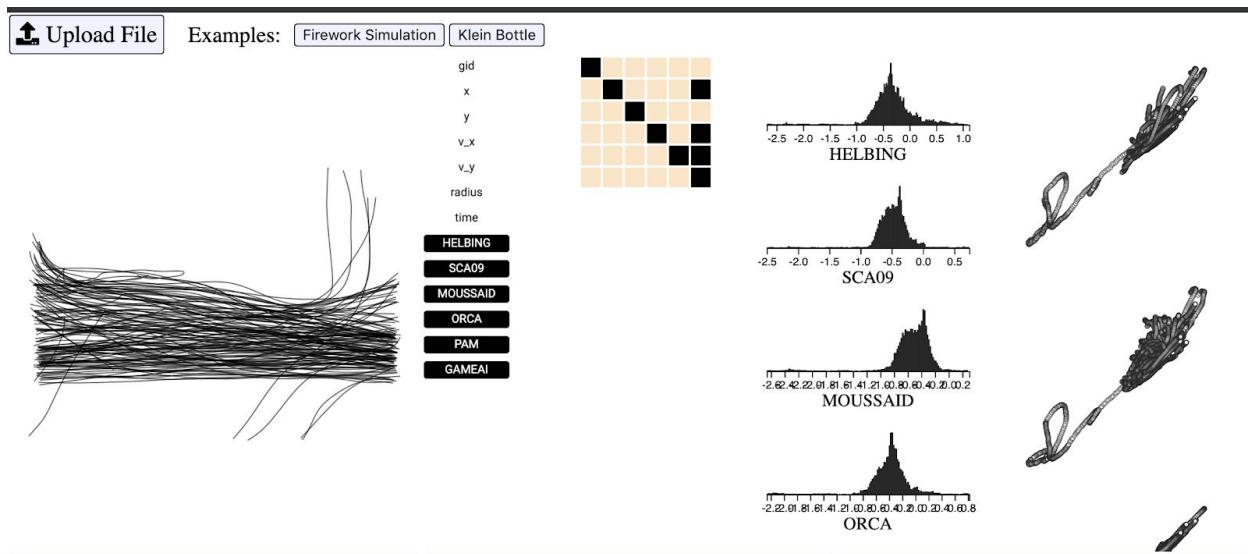
- I'm thinking of combining the scatterplots and distributions into one widget.
 - This widget is also something that could benefit from view config settings.
 - Look into using provenance library
- Layout

- Maybe could specify frames as “fit-to-content” and shrink to best aspect ratio given direction of container.
- Coloring along paths
 - <https://github.com/mnsht/gradient-path>
- Cell Project
 - Relevant reading?
 - Slack reminder

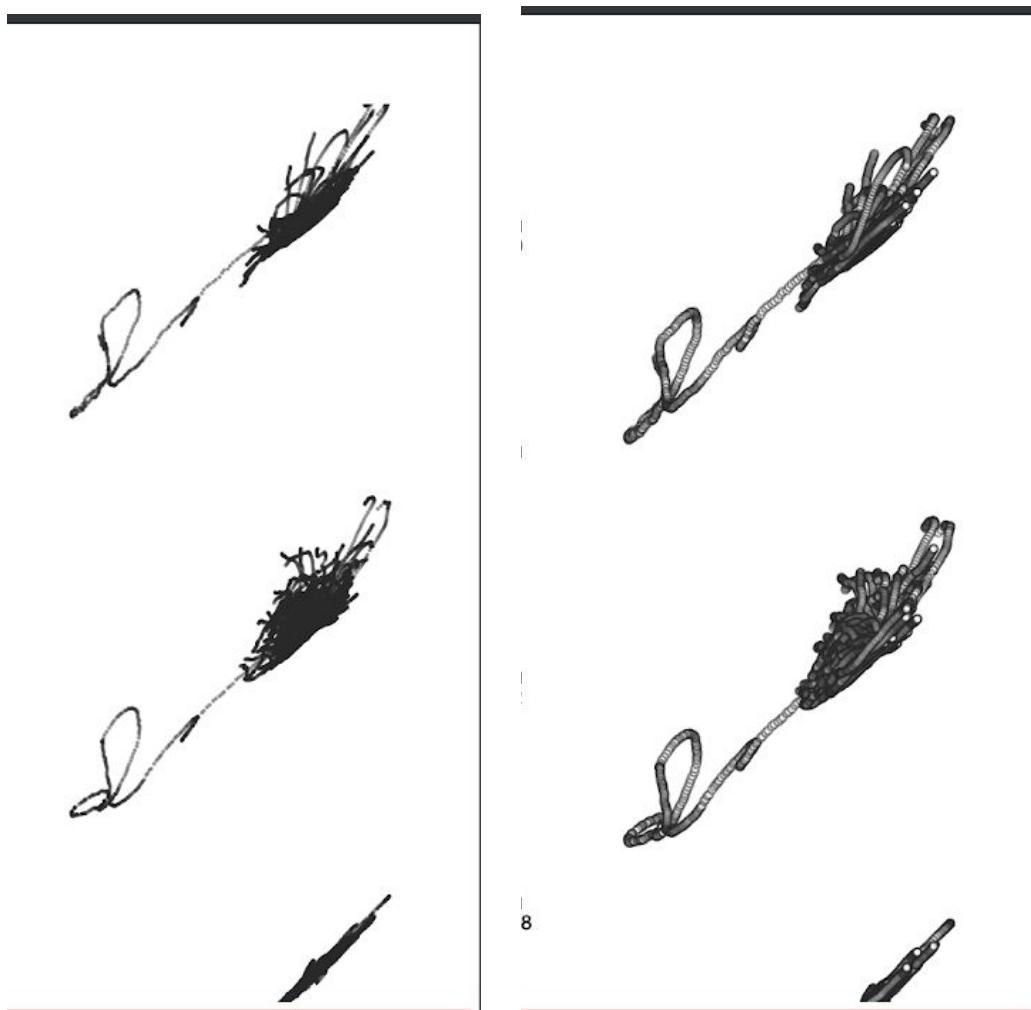
Updates on Histograms and Scatterplots:



Another option for point rendering:

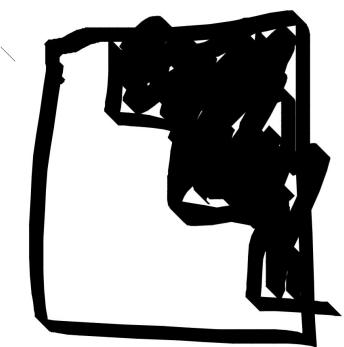


Not sure which is better, but the second option did give an interesting and gave a 3D look that I was not expecting.



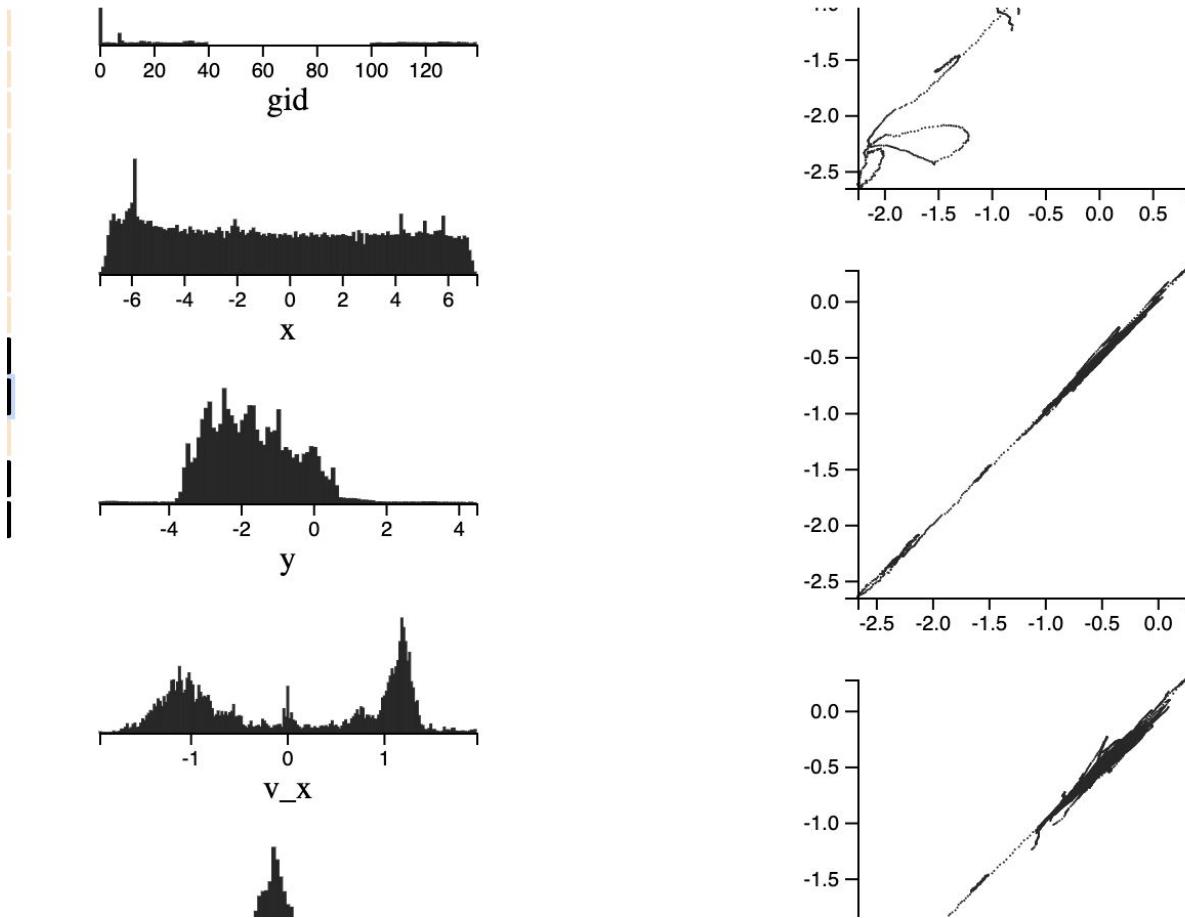
Another important performance consideration is opacity. Having opacity less than one can increase rendering time. I do think transparency is nice, but for now, I will try out full opacity and radius of 0.5

Random thought. For the matrix select widget, I sometimes don't know what I want, so I select a bunch, which is laborious. I would use a half-select all button.



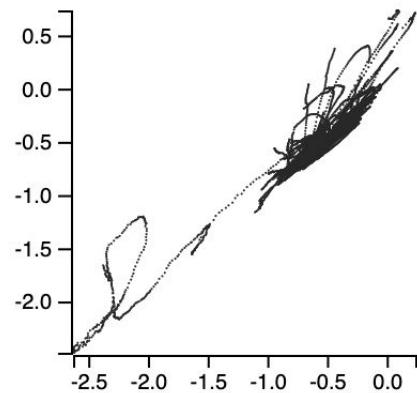
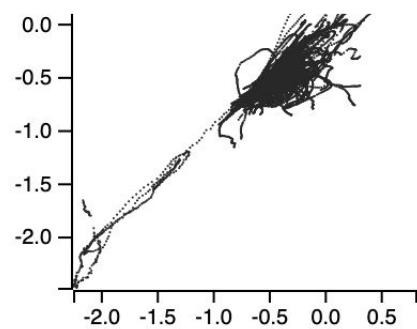
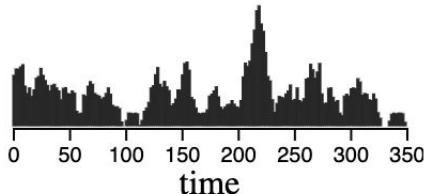
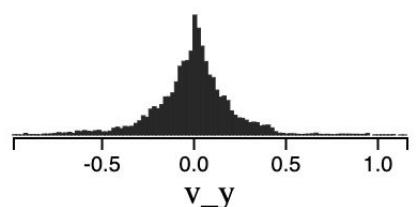
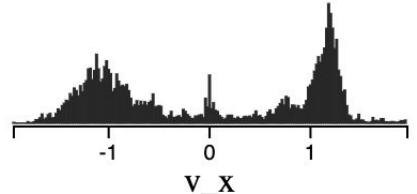
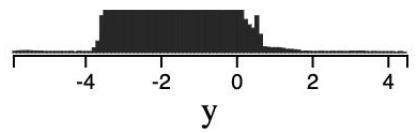
That selects the top half of the matrix, could consider having the opposite half as well.

I hardcoded number of ticks so there isn't much chance of label overlap:

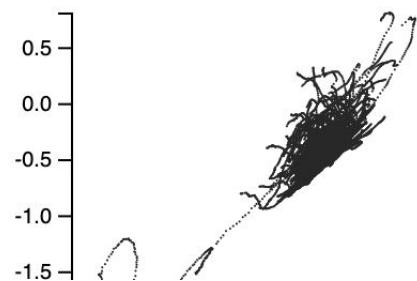
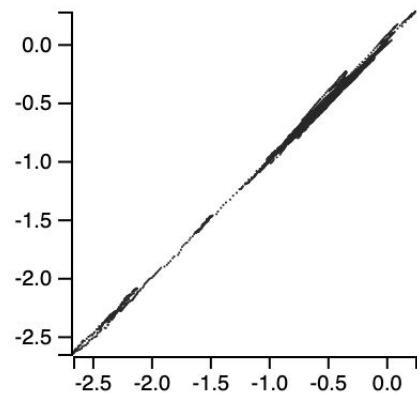
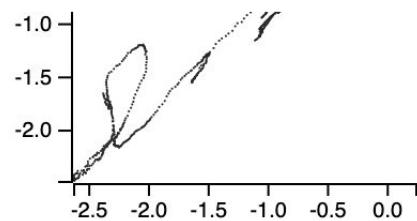
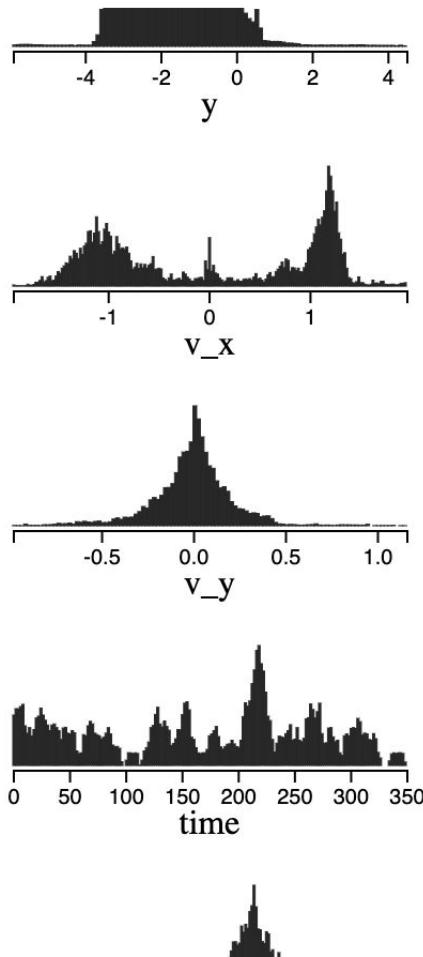


Looking at this is making me start to reevaluate if I want padding between label and data extent at all. Let's see what it would look like:

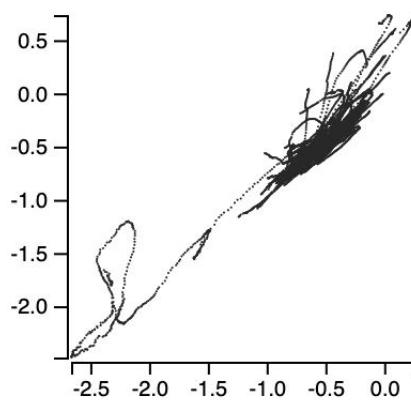
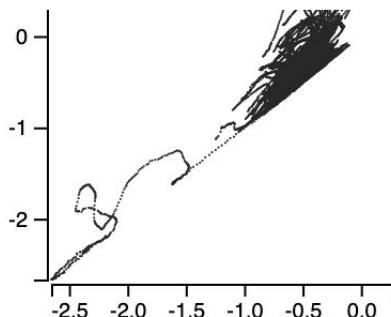
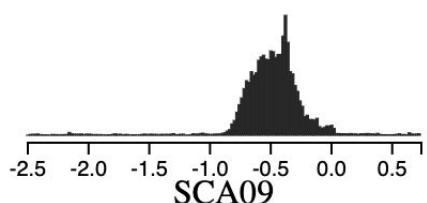
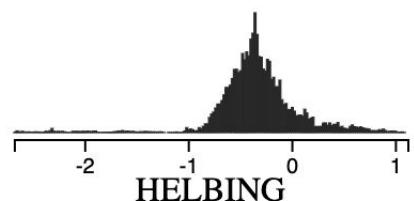
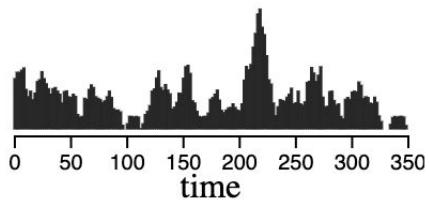
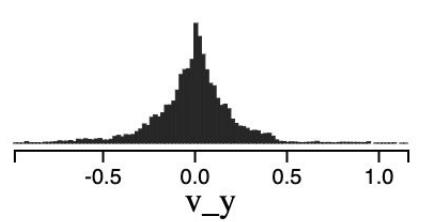
Padding = 1



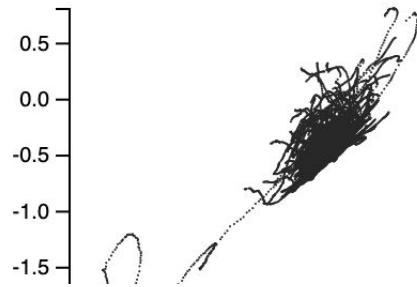
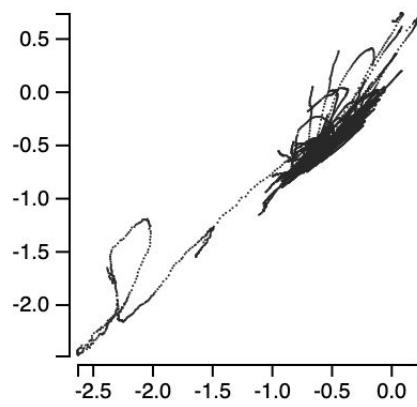
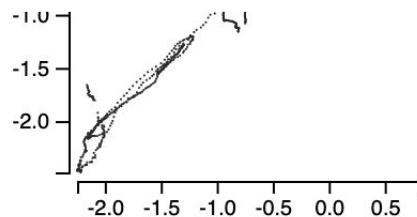
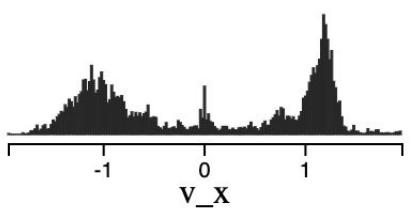
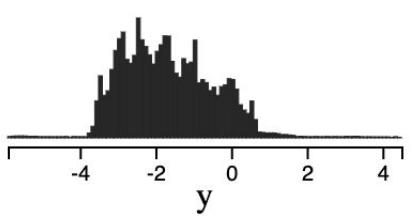
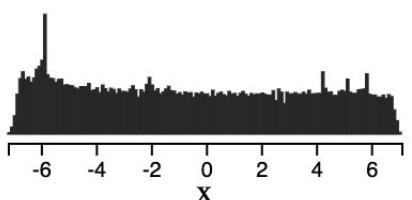
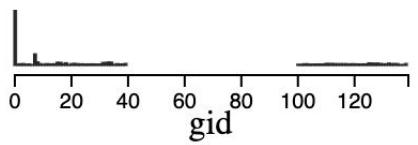
Padding = 2



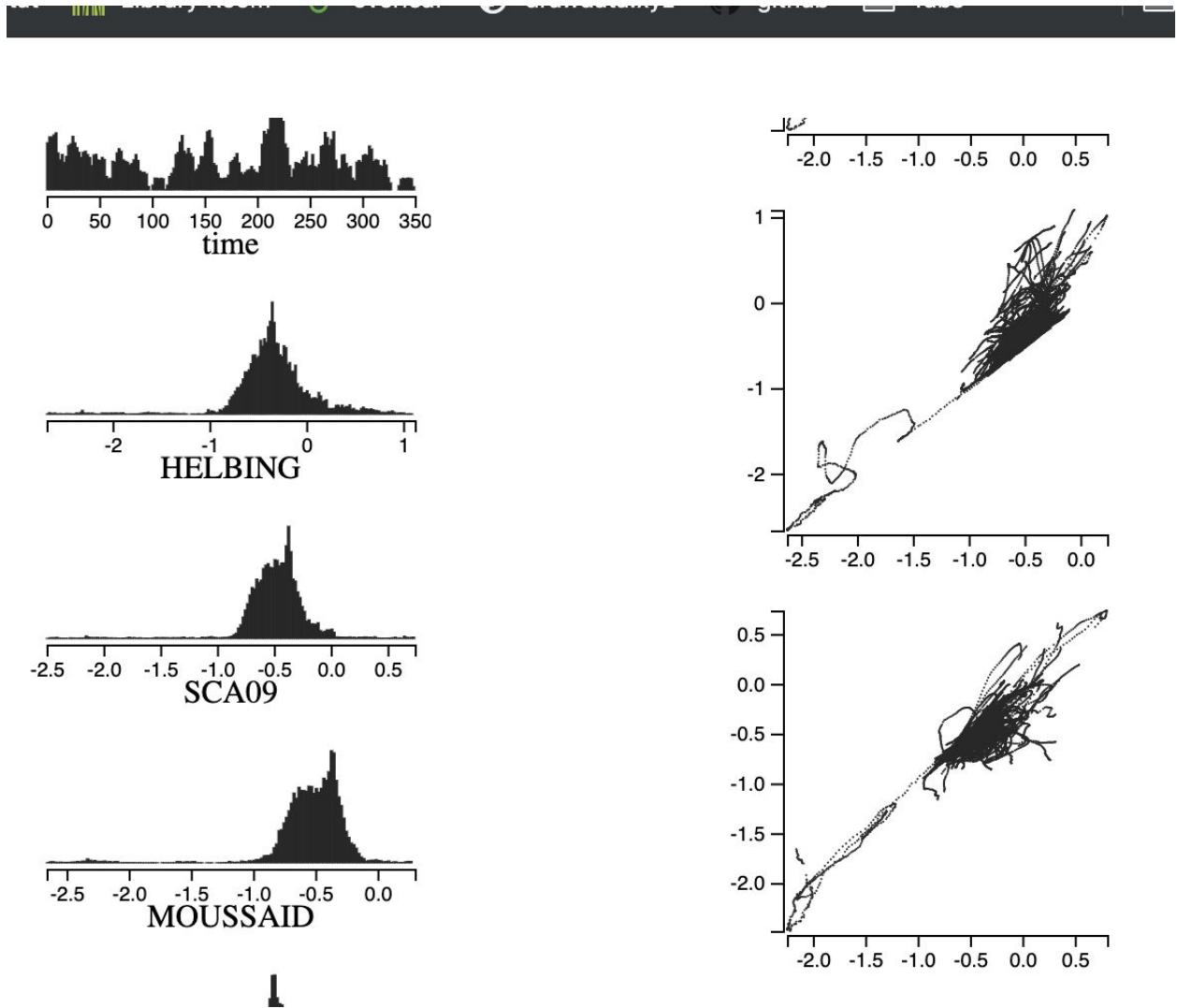
Padding = 3



Padding = 4

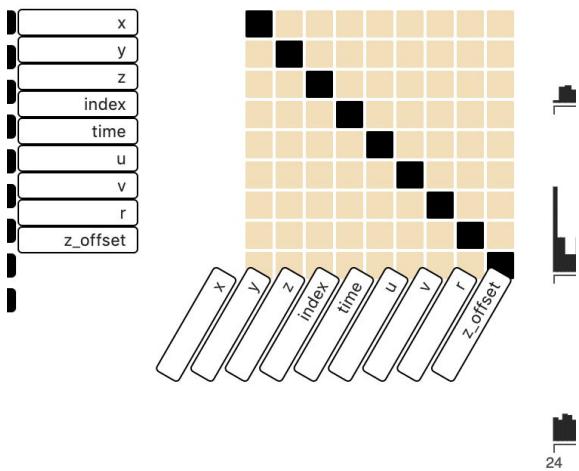


Padding = 5



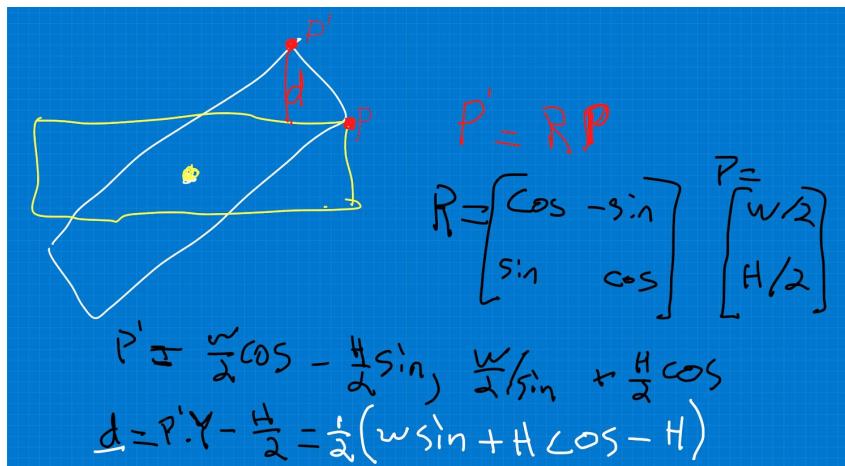
I think these do help, especially for the histogram, not as sure about the scatterplot. I do personally prefer a more compact view, so my favorite of these is with padding=2.

For the tilted buttons the way I got it to work was with absolute positioning, and css transforms. This requires a little math. In this screenshot I have the horizontal offset pretty much worked out, but the vertical offset is not right.

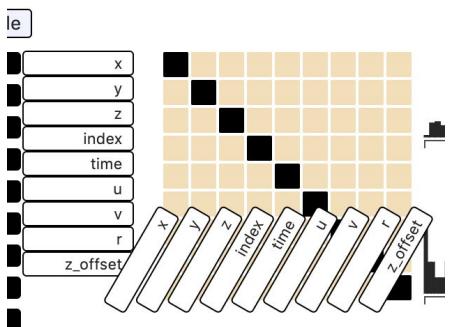


24

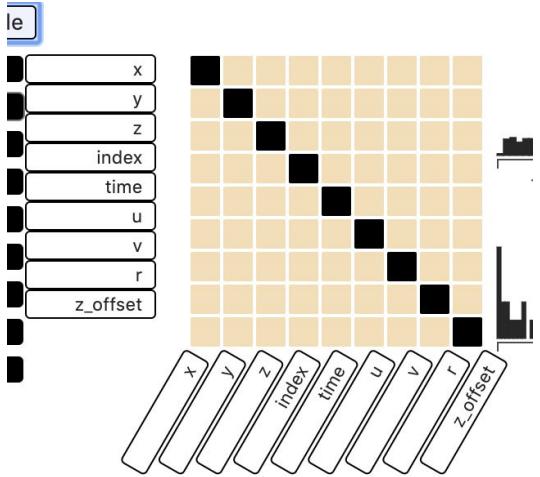
Here is my math for figuring out the vertical offset.



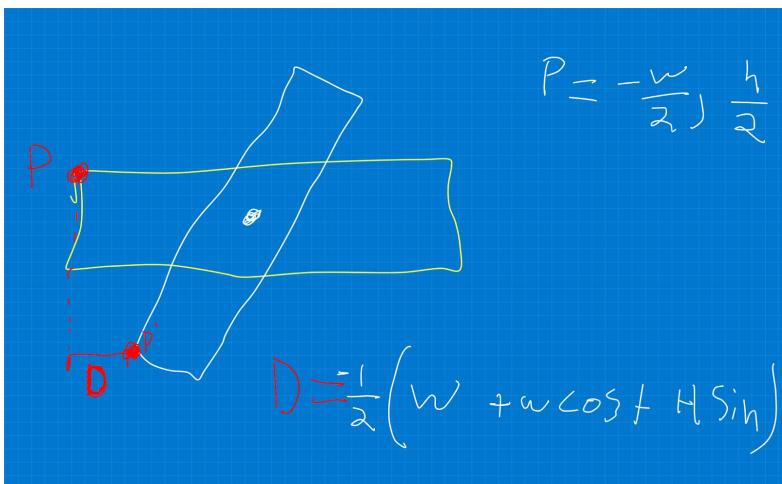
Nailed it...



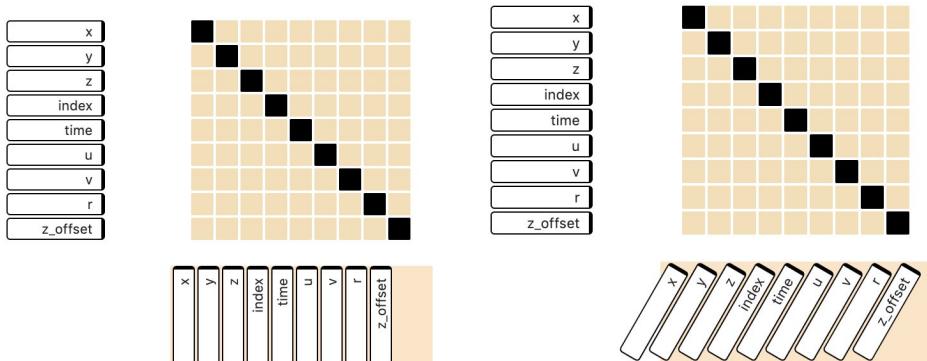
I figured out that convention for rotation direction for this equation was opposite for rotation transforms in CSS. After flipping the sign... Success!



A Similar process can be done for the horizontal offset:

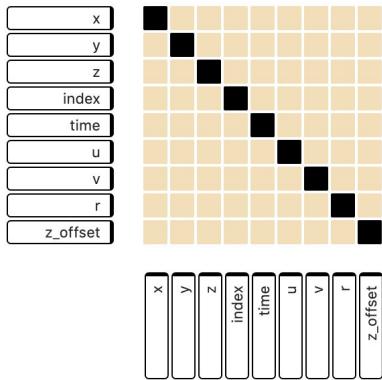


Success, now the leftmost button is anchored at the top left of the container, no matter the angle:

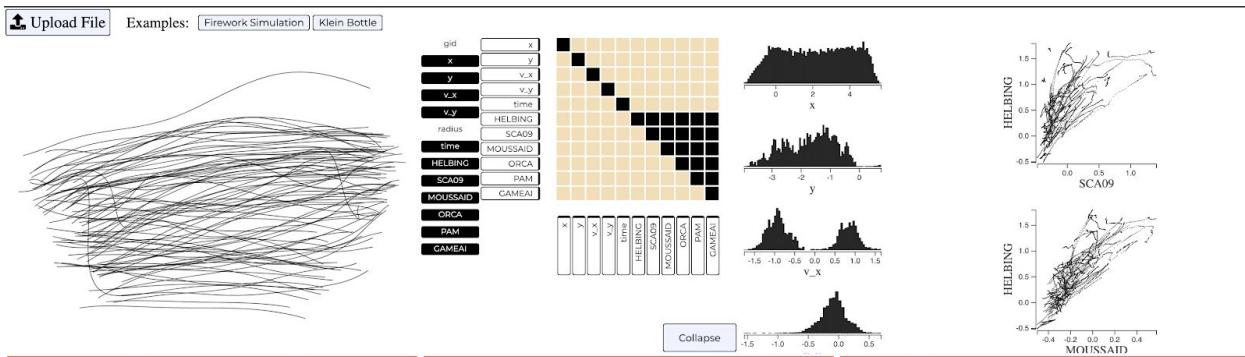


I would like to use the angled buttons, with some line connecting to the matrix, but for now since the vertical buttons line up, I will use that for the short term. This is not a long term solution

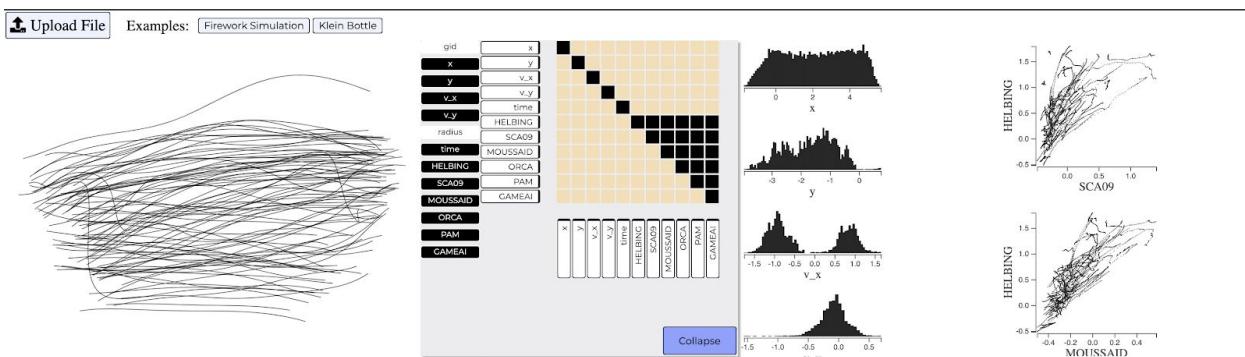
though. It only happened to line up since I serendipitously selected the same height and padding for the buttons as the matrix cells.

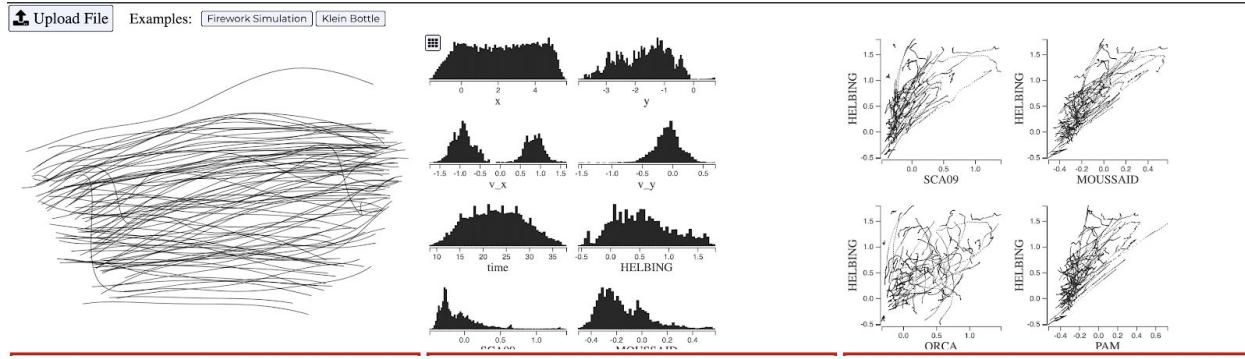


Since this widget takes up a decent amount of space I added the ability to collapse it.



With mouse hovered over collapse button:





I'm quite happy with this. I'm not sure if someone would discover this feature on their own, but I do think it would be easy to use if shown once.

In terms of features I completed all of the proposed must have features as well as two of the stretch goal features.

Must Have Features

- ~~The base programming framework must be laid so that programmatically adding new widgets in the future is easy.~~
- ~~2 linked widgets must be completed~~
 - ~~A plot of paths traced by all trajectories~~
 - ~~Metric distributions of points with brushing capability~~

Optional Features

Stretch Goal - Tier 1 - My goal is to finish these, but will cut if they cannot be finished in time.

- ~~The plot of all paths will also support animation of Trajectories~~
- 2 additional widgets
 - Level of detail plot with brushing capability
 - Detailed table

Stretch Goals - Tier 2 - I might get started on these, but likely will not get far into this list before project submission.

- Metric console
- Small multiple path subdivision
- ~~Small multiple scatterplots~~
- Support metrics on trajectories and groups
- Add some number of prebuilt metrics (e.g. speed, acceleration, curve depth)