

# Visualizing Neuron Electrophysiology via NeuroElectro.org

## Report: Update 1

November 26, 2015

Authors:

Michael Gottlieb [mikemgottlieb@gmail.com](mailto:mikemgottlieb@gmail.com)

Emily Hindalong [ehindalong@gmail.com](mailto:ehindalong@gmail.com)

Dmitry Tebaykin [dmitry.tebaykin@gmail.com](mailto:dmitry.tebaykin@gmail.com)

# Previous work

## Overview

Neuroelectro is a Django text mining application [<http://neuroelectro.org/>] and was published as an article in Frontiers in Neuroinformatics [1]. Through the combined use of text mining and manual curation, it currently hosts experimental data from over 500 articles and is expected to grow to host the experimental data of about 5,000 articles. The data of each article can be accessed by type of neuron, electrophysiological property, or via a table of articles.

Neuroelectro is unique amongst text mining projects in that it allows end users to interact with curated data directly. Most text-mining tools in the biomedical domain assume that the end user will want an association matrix for terms in a controlled vocabulary, such as MEDLINE or MeSH terms [26,27]. These tools automatically generate and output an association matrix without providing the user with a way to interface with the primary data. This limits the analyses the user can perform.

Furthermore, Neuroelectro provides some crude data visualizations in the form of static scatterplots and a single PCA analysis plot. Most text-mining tools in biomedicine do not use visualization at all, and those that do are restricted to analyses on the derived association matrix. For example, VOSviewer [2] uses colour and spatial position to visualize the semantic clustering and strength of association across text mined terms. The Trading Consequence project [3] focuses on mined trading documents supported by controlled vocabularies to generate maps of commodity trading over time.

## Solutions to Similar Problems

### **Exploring relationships (abstract task 1 in proposal)**

Exploration of relationships between the different properties in Neuroelectro's dataset is supported by the current version of Neuroelectro. However, it is a static set of strip plots that do not account for all combinations of variables and do not allow any interaction.

Exploration of relationships is a common task in many analytics platforms such as Tableau [17], SAP BOBJ AOLAP [32] and Microsoft Excel [33]. Generally, these platforms provide a tabular view of the data in addition to customizable visualizations to enhance users' exploration of the data. Our goal differs from these platforms as we are not including a tabular view, we are

limiting the users' choices to provide a simpler experience, and we are using plots that are not easily achieved with these platforms (e.g. interactive plots, hive-plots).

### **Providing an overview of data (abstract tasks 3 and 4 in proposal)**

Essentially, we are facing a problem of visualizing a network when we are trying to give an overview of our data. Over the years many solutions have been proposed for this type of task: hairball [35], matrix [34], arc diagram [31], call network [37], hive plot [29], etc. Simply visualizing the network as a collection of nodes connected with edges (the hairball approach) seems impractical due to a large number of nodes and connections (currently: 150 nodes and ~10k edges), scaling is also a problem since the hairball only gets bigger with time. The matrix approach deserves some credit in terms of data visibility and it is a familiar visualization style to biologists, but there are 2 issues with utilizing matrices for this task: 1) Our data is 3 dimensional (neuron type, ephys. property, metadata) and 3D matrices are usually very hard to interpret, we could provide a faceted view of 1 matrix per metadata as a possible solutions, but the amount of screen space that would require is enormous; 2) Matrices do not scale well, the labels get too small to be legible at some point. A call network visualization would end up looking very similar to a hairball in our case, as a result we had to discard this possibility due to scalability issues. Arc diagrams came in as a close second as our visualization of choice - they are easy to interpret, pleasant to look at and they can scale reasonably well with the amount of evidence in the database. The problem with arc diagrams is that all nodes would end up being on one line and that does not represent the 3 distinct groups of nodes (neuron types, ephys. properties, metadata) in our data.

As a result, we decided to use hive plots for providing an overview of our data. Krzywinski, Birol, Jones and Marra [30] describe the advantages of hive plots in terms of gaining quantitative understanding when visualizing networks. They also support: multiple axes, information encoding in the nodes and edges, scaling. The one issue with hive plots is that they are a fairly new visualization style and researchers may have trouble understanding what they are looking at. However, we plan to provide a guide to interpreting the hive plot as well as provide links to the supporting literature. This feature would be on-demand and can be disabled in the application preferences.

## Applications of Similar Solutions

### **Filter panels**

The filter panel paradigm, where one panel is used to control what data appears in the main panel, is well established in visualization domain [24,25,28]. An alternative solution is the *filter bar*, which uses less screen real estate [22]. However, we have opted to stick with filter panels because they will never interfere with the main view and will make it easier for the user to track which filters are applied at any given time. Furthermore, the number of filtering options that we offer will require a larger section of the screen.

There are two basic attribute-based filtering paradigms: drill-down and parallel selection [21]. As the referenced blog post describes, Amazon uses drill-down filtering and Kayak uses parallel filtering. Our solution uses a hybrid of these, allowing the user to drill-down categories and apply parallel selection within. We intend to refer to this blog post when designing the specific details of our filter panel.

### **Connected scatterplots**

Since Neuroelectro data is rather diverse (dozens of electrophysiology properties for each of over one hundred neuron types), we plan to utilize scatterplots [16] and connected scatterplots [7] for answering research-oriented questions. Haroz *et al.* [7] showed the effectiveness of the latter in representing time-series data: even though connected scatterplots are novel to many users, they are excellent at being intuitive to understand and capturing and holding the viewer's attention.

### **Linked highlighting**

There are a number of interaction approaches to linked highlighting in scatter plots [23]. Through our consultation with the stakeholder, linked highlighting on hover was emphasized as a critical element. However, this not the only means of linked highlighting available. For example, linked *brushing*, where the user selects a subset of points to be highlighted, is a popular choice for multi-selection [23, 36]. Further conversations with our stakeholder as we gain more exposure will allow us to finalize our design decisions.

### **Hive plot**

Overviews of the data will make use of the work done by Krzywinski *et al.* [5] and Hanson [9] for our proposed hive plot. Various good examples of hive plot visualizations of networks are shown on the hive plot website [29]. We will be developing our visualization based on the features that are most meaningful for our data and the questions we are trying to answer with this view (sparsity of data, node degree, node centrality and reachability, outliers and general trends in the data).

### **Colour**

We have decided to use colour as a channel for linked-highlighting. Previous research suggests that colour is one of the most powerful visualization channels [20, 8] and, when used correctly, it can provide insight into the data so intuitively that the user wouldn't even need a legend to understand what kind of data the channel encodes for.

## Implementation Solutions

We will likely take advantage of the work done in interactive visualizations such as D3.js [4] and AngularJS [14] or RShiny [15]. Although D3.js and Angular.js are arguably more powerful in terms of possible customizations and optimizations than RShiny, both of them would require a steep learning curve and time input we do not have. In the interest of time and given the team's expertise in R, we are leaning towards using RShiny and will most likely take advantage of previous work available via github and the Shiny website. For example, we have found solutions to set up multiple plots [10], use linked views [11], filter data [12], and rearrange plot layout [13]. The hive plot library HiveR.R developed by Hanson [9] provides us with the tools we need to create an interactive representation of the whole dataset.

## Bibliography

1. Tripathy, Shreejoy J., et al. "NeuroElectro: a window to the world's neuron electrophysiology data." *Frontiers in neuroinformatics* 8 (2014).
2. Van Eck, Nees Jan, and Ludo Waltman. "Text mining and visualization using VOSviewer." *arXiv preprint arXiv:1109.2058* (2011).
3. Hinrichs, Uta, et al. "Trading Consequences: A Case Study of Combining Text Mining and Visualization to Facilitate Document Exploration." *Digital Scholarship in the Humanities* (2015).
4. Bostock, Michael, Vadim Ogievetsky, and Jeffrey Heer. "D<sup>3</sup> data-driven documents." *Visualization and Computer Graphics, IEEE Transactions on* 17.12 (2011): 2301-2309.
5. Krzywinski, Martin, et al. "Hive plots—rational approach to visualizing networks." *Briefings in bioinformatics* 13.5 (2012): 627-644.
6. Rieder, Christian, et al. "Interactive visualization of multimodal volume data for neurosurgical tumor treatment." *Computer Graphics Forum*. Vol. 27. No. 3. Blackwell Publishing Ltd, (2008).
7. Steve Haroz, Robert Kosara, Steven L. Franconeri. "The Connected Scatterplot for Presenting Paired Time Series." *Transactions on Visualization and Computer Graphics*, (2016): <https://research.tableau.com/sites/default/files/Haroz-TVCG-2016.pdf>
8. Vidya Setlur, Maureen C. Stone. "A Linguistic Approach to Categorical Color Assignment for Data Visualization" *The IEEE Information Visualization Conference* (Chicago, October 25-30, 2015): [https://research.tableau.com/sites/default/files/setlurstoneinfovis2015\\_2.pdf](https://research.tableau.com/sites/default/files/setlurstoneinfovis2015_2.pdf)
9. Bryan A. Hanson. "HiveR: 2D and 3D Hive Plots for R." (2015): <academic.depauw.edu/~hanson/HiveR/HiveR.html>
10. <https://gist.github.com/wch/5436415>
11. <http://shiny.rstudio.com/gallery/plot-interaction-zoom.html>
12. <http://shiny.rstudio.com/gallery/basic-datatable.html>
13. <http://shiny.rstudio.com/articles/layout-guide.html>
14. <https://material.angularjs.org/latest/>
15. <http://shiny.rstudio.com/>
16. Michael Friendly, Daniel Denis. "The early origins and development of the scatterplot" *Journal of the History of the Behavioral Sciences* (2005): Vol. 41(2), 103–130
17. <http://www.tableau.com/>
18. <http://www.tableau.com/gartner-magic-quadrant-2015>
19. <http://datablick.com/2015/04/13/circular-and-hive-plot-network-graphing-in-tableau-by-chris-demarti>  
[ni/](http://datablick.com/2015/04/13/circular-and-hive-plot-network-graphing-in-tableau-by-chris-demarti)
20. Tamara Munzner. "Visualization Analysis and Design" (2014) A K Peters/CRC Press, Print ISBN: 978-1-4665-0891-0, eBook ISBN: 978-1-4665-0893-4
21. <http://www.uxmatters.com/mt/archives/2009/09/best-practices-for-designing-faceted-search-filters.php/>
22. <http://www.uxforthemasses.com/filter-bars/>
23. [https://www.perceptualedge.com/articles/visual\\_business\\_intelligence/coordinated\\_highlighting\\_in\\_context.pdf](https://www.perceptualedge.com/articles/visual_business_intelligence/coordinated_highlighting_in_context.pdf)
24. Bearavolu, Ratna, et al. "A visualization tool for situational awareness of tactical and strategic security events on large and complex computer networks." *Military Communications Conference, 2003. MILCOM'03. 2003 IEEE*. Vol. 2. IEEE, 2003.,
25. Dumas, Maxime, Jean-Marc Robert, and Michael J. McGuffin. "Alertwheel: radial bipartite graph visualization applied to intrusion detection system alerts." *Network, IEEE* 26.6 (2012): 12-18.
26. French, Leon, and Paul Pavlidis. "Informatics in neuroscience." *Briefings in bioinformatics* 8.6 (2007): 446-456.

27. Rebholz-Schuhmann, Dietrich, Anika Oellrich, and Robert Hoehndorf. "Text-mining solutions for biomedical research: enabling integrative biology." *Nature Reviews Genetics* 13.12 (2012): 829-839.
28. Sopan, Awal, et al. "Community Health Map: A geospatial and multivariate data visualization tool for public health datasets." *Government Information Quarterly* 29.2 (2012): 223-234.
29. hive plot website: [www.hiveplot.net](http://www.hiveplot.net)
30. Krzywinski M, Birol I, Jones S, Marra M. "Hive Plots — Rational Approach to Visualizing Networks." *Briefings in Bioinformatics*, doi: 10.1093/bib/bbr069
31. Arc diagrams in R (network visualization): <http://gastonsanchez.com/blog/got-plot/how-to/2013/02/02/Arc-Diagrams-in-R-Les-Miserables.html>
32. <http://help.sap.com/boaolap40>
33. <https://products.office.com/en-ca/excel>
34. Henry N, Fekete JD. Matrixexplorer: a dual-representation system to explore social networks. *IEEE Trans Visu Comput Graph* (2006). 12(5):677-84
35. Aitaluk M, Sedova M, Ray A, et al. "Biological Networks: visualization and analysis tool for systems biology." *Nucleic Acids Res* (2006). 34:W466-71
36. [https://github.com/rstudio/ggvis/blob/master/demo/rmarkdown/linked\\_brush.Rmd](https://github.com/rstudio/ggvis/blob/master/demo/rmarkdown/linked_brush.Rmd)
37. Yan KK, Fang G, Bhardwaj N, Alexander RP, Gerstein M. 2010. "Comparing genomes to computer operating systems in terms of the topology and evolution of their regulatory control networks." *Proc Natl Acad Sci USA*. 107(20): 9186-9191

# Progress Update

## 1. Priorities and High-Level Design Changes

We held another meeting with our stakeholder shortly after submitting our proposal to discuss priorities and ask questions.

He was very pleased with and excited about the plots view and filter menu (components 1 and 4 in our mock-up). He encouraged us to focus on getting these components working well as a first priority. It was agreed that the hive-plot would be a useful but lower priority item.

During our discussion, it became clear that our design for the brain view (component 2) had not captured the stakeholder's vision. In fact, the stakeholder realized he was not certain of his vision for this component. Ideally, it would express something about the electrophysiological characteristics of certain brain regions, and this would require some form of aggregation based on properties of different neuron types present in each region. This is a considerably more sophisticated problem than the one required by our original design. As such, we have decided to table the brain view for now and focus on the other views.

Our stakeholder has also offered to provide us with two additional pieces of information: (1) A hierarchy of neuron type categories for a more compact tree view representation and (2) for each quantitative attribute, a range and (if discrete) a step size. This will inform the slider designs.

## 2. Specific Design Changes and Implementation Progress

### **Overall layout**

The overall layout is similar to what we originally proposed. We currently have the control panel and a plots view (components 1 and 4 in the mock-up). We are not certain where we will place the hive-plot view. Also, R Shiny does not appear to support resizable panels, so we will likely have to fix the sizes.

### **Plots view**

We decided to impose a static grid with room for 4 plots. When the page is first loaded, the regions will be populated with a default plot set to orient the user.

Rather than having a separate pop-up to define the axes, we have a button that toggles between a plots view and a change axes view.



We have made considerable progress on the display of plots, including axis label orientation and placement. We have implemented tooltips that appear when the mouse hovers over a point.

Additionally, we have implemented linked-highlighting and added a widget that allows the user to choose whether to highlight on hover or click. Highlight on hover causes a point to turn blue and double in diameter when it is hovered-over. If that data point is present in the other plots, it will turn blue and double in diameter there as well. Highlight on click allows the user to select multiple points for highlighting within a single plot by clicking them. A clear highlighting button allows them to remove this highlighting.

We have also added an option that allows the user to remove points from plots by clicking on them (which can be useful in the presence of outliers). This overrides highlight on click. We are still considering ways to make this less confusing. The layout of the control widgets is also a work in progress.

## **Control panel**

Progress on the control panel has been slow because R Shiny does not provide a tree widget. We have experimented with two alternatives: embedding an html widget and building a widget from scratch using Shiny components such as conditional panels and buttons. The first approach was not feasible because we would like to embed Shiny widgets (checkbox groups and slider bars) in the tree. We have made some progress using the second approach - we can now populate, expand, and collapse a tree, but we have not yet added the filter control widgets. However, we recently discovered the shinyTree package. Depending on how flexible it is, we may be able to adapt this to our purposes instead.

## **Hive plot view**

We have created a hive plot using hiveR package and a couple of libraries that extend it. At the moment, the hive plot shows the presence of evidence lines for each pair of type: neuron type - ephys. property, neuron type - metadata, ephys. property - metadata (work in progress). We are exploring the possibilities of including various features into the plot: using transparency for indicating the edge weights (number of evidence lines per connection), grouping neuron types by brain region since 109 individual neuron types is too difficult to distinguish on a single axis, and allowing the user to click on the nodes to display more information about each point. Figure 7 shows the current version of the hive plot. The axes are sorted alphabetically, each node's size is proportional to its betweenness centrality, and each node's color is proportional to its number of connections.

There are a number of nice-to-have interactivity features that we would like to add to the hive plot. That includes the ability to filter data in the analysis view based on selected neuron types / ephys. properties and metadata.

## Revisiting Unresolved Design Questions

1. Should we use a static grid in the plots view or resize based on the number of plots in the display?
  - a. We will use a static grid of size 4.
2. Should axis ranges in the plots view reflect the full range for an attribute or the filtered range?
  - a. The axis ranges will be dynamic. This is ggvis's default behaviour.
3. Should the user be allowed to put categorical attributes on both plot axes? If so, how should we show overlapping points?
  - a. Yes. We have not yet resolved the overlapping points problem. The library we are using (ggvis) does not support "jitter." We are considering alternative representations, such as frequency heatmaps, for plots with categorical attributes on both axes. This doesn't solve the overlapping points problem in general. Points may still overlap when one axis is categorical and the other quantitative or (to a lesser extent) when both axes are quantitative.
4. How many neuron types should we restrict the brain view to? How do we select which ones go on the view if more neuron types are selected?
  - a. We have decided not to implement the brain view (for now).
5. Should all values in the control panel be ticked or unticked by default?
  - a. Ticked. That way the user starts with an overview of all the data.
6. Should we use a selection widget that is more scalable than tick boxes?
  - a. We have decided that tick boxes are a viable solution. There are currently over a hundred neuron types, but we can improve tree navigation by grouping neuron types under subnodes according to a predefined hierarchy (to be provided by stakeholder). The other categorical attributes (e.g. species) do not have prohibitively many values. The number of values for these attributes is not likely to increase much in the future (as confirmed by our stakeholder).

# Appendix: Current UI

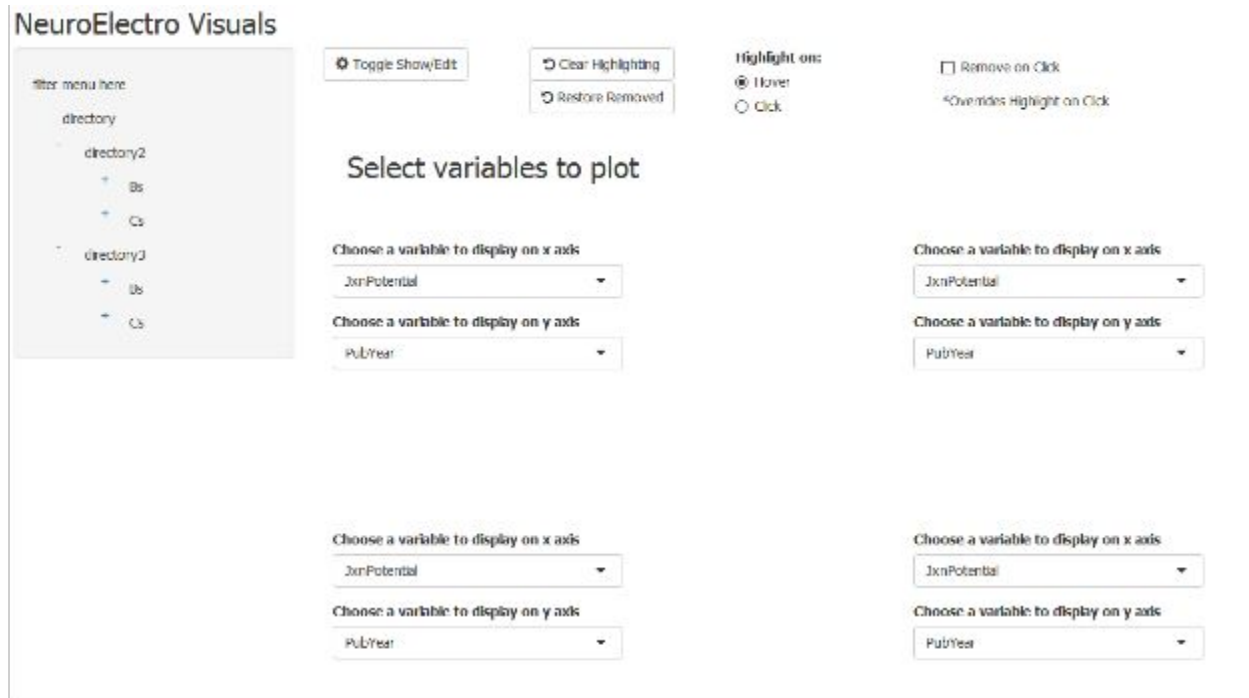


Figure 0. Entry point to application. Filter menu on left hand panel, main view with axis selection and plots on right hand side.

**Highlight on:**  
 Hover  
 Click

Remove on Click  
\*Overrides Highlight on Click

## Select variables to plot

Choose a variable to display on x axis

Species

- Pmid
- PubYear
- TableID
- Species
- ElectrodeType
- PrepType
- JxnPotential
- JxnOffset

Choose a variable to display on x axis

JxnOffset

Choose a variable to display on y axis

RecTemp

Choose a variable to display on x axis

AHP.amplitude

Choose a variable to display on y axis

external\_0\_K

Choose a variable to display on x axis

access.resistance

Choose a variable to display on y axis

PubYear

Figure 1. Axis selection view.



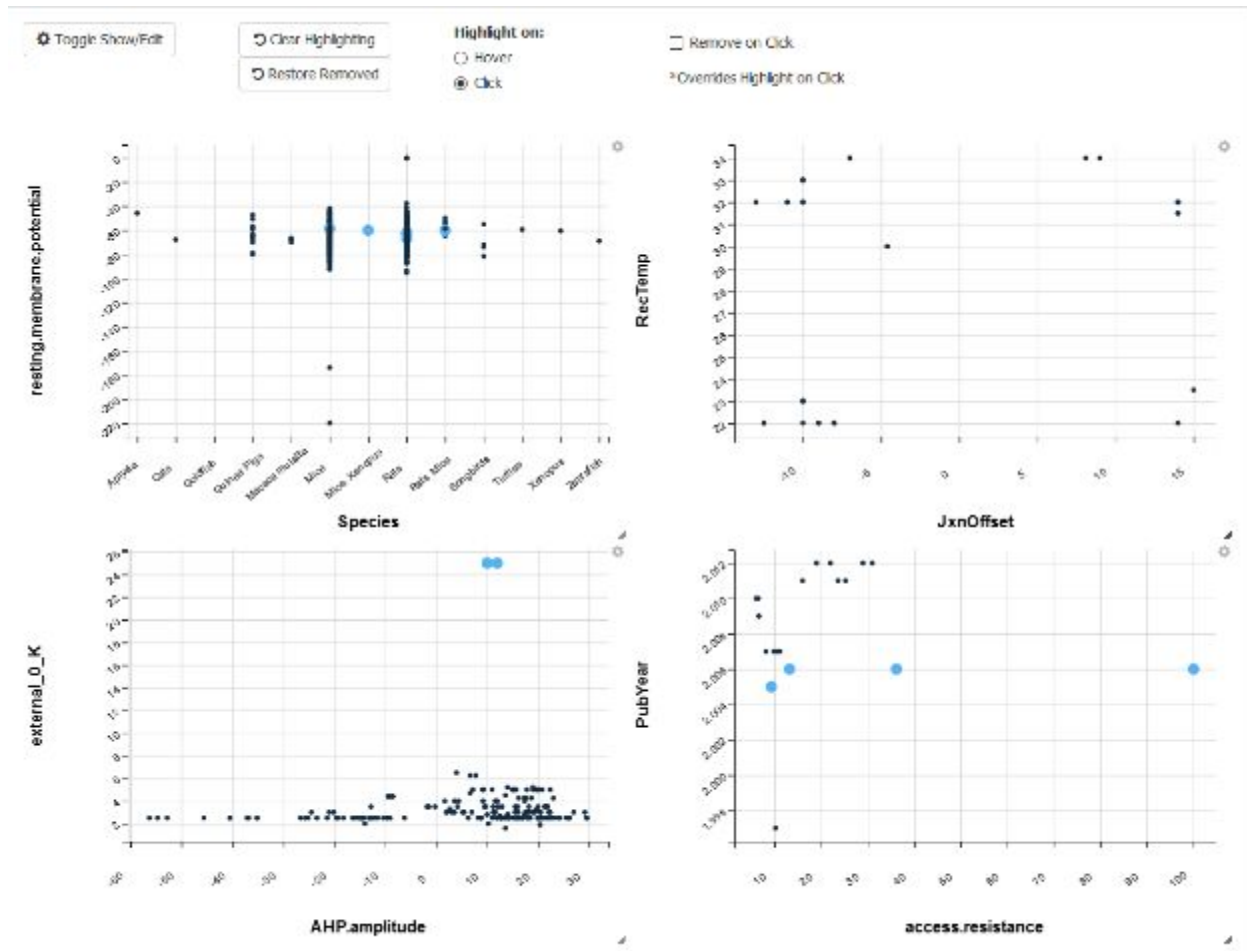


Figure 3. Switching to click mode in top bar allows highlighting of many points at once.

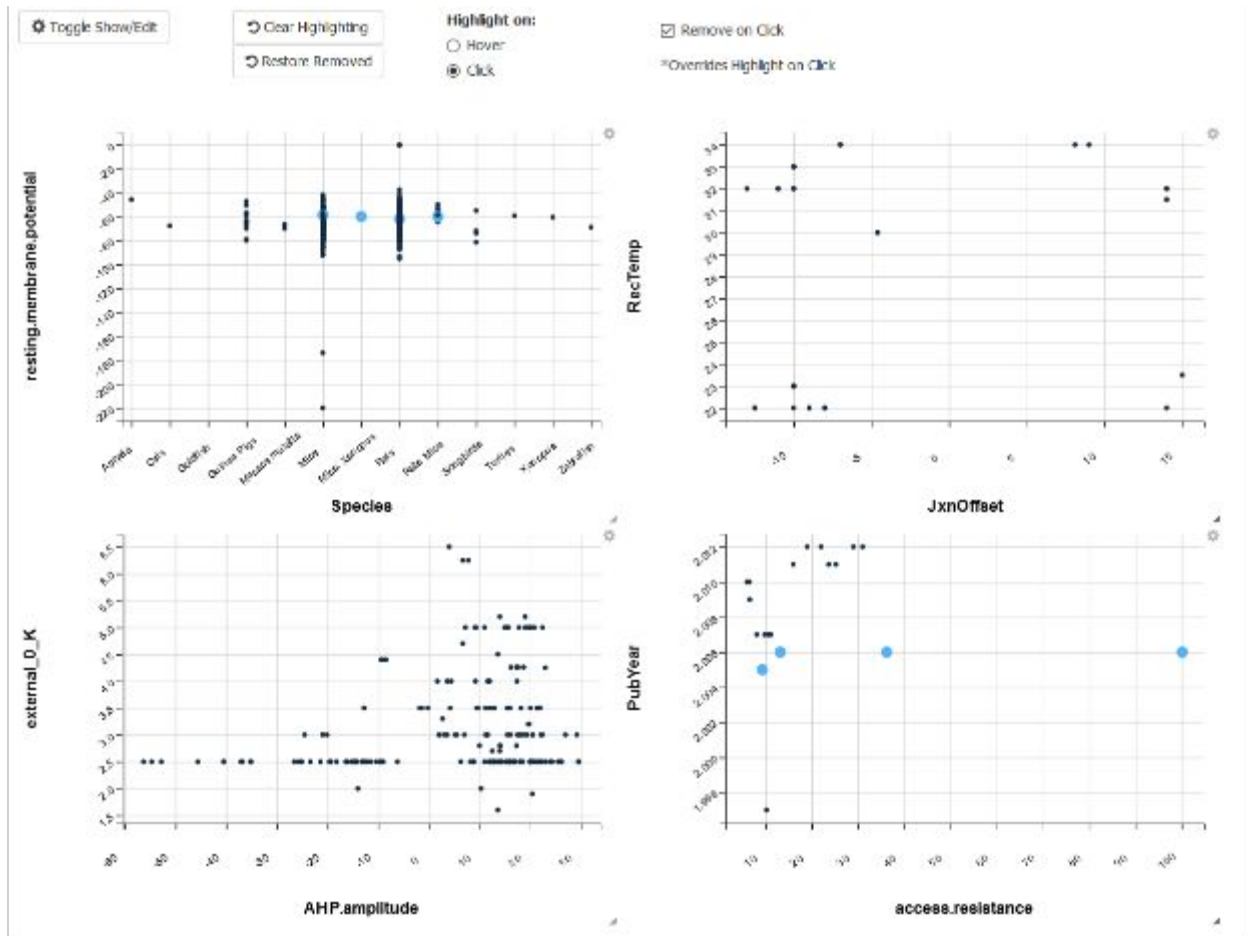


Figure 4. Remove on click mode activated. In bottom left plot two points, considered outliers by our stakeholder, have been clicked on and thus removed from all plots in which they appear. This causes the bottom left plot's axes to resize.

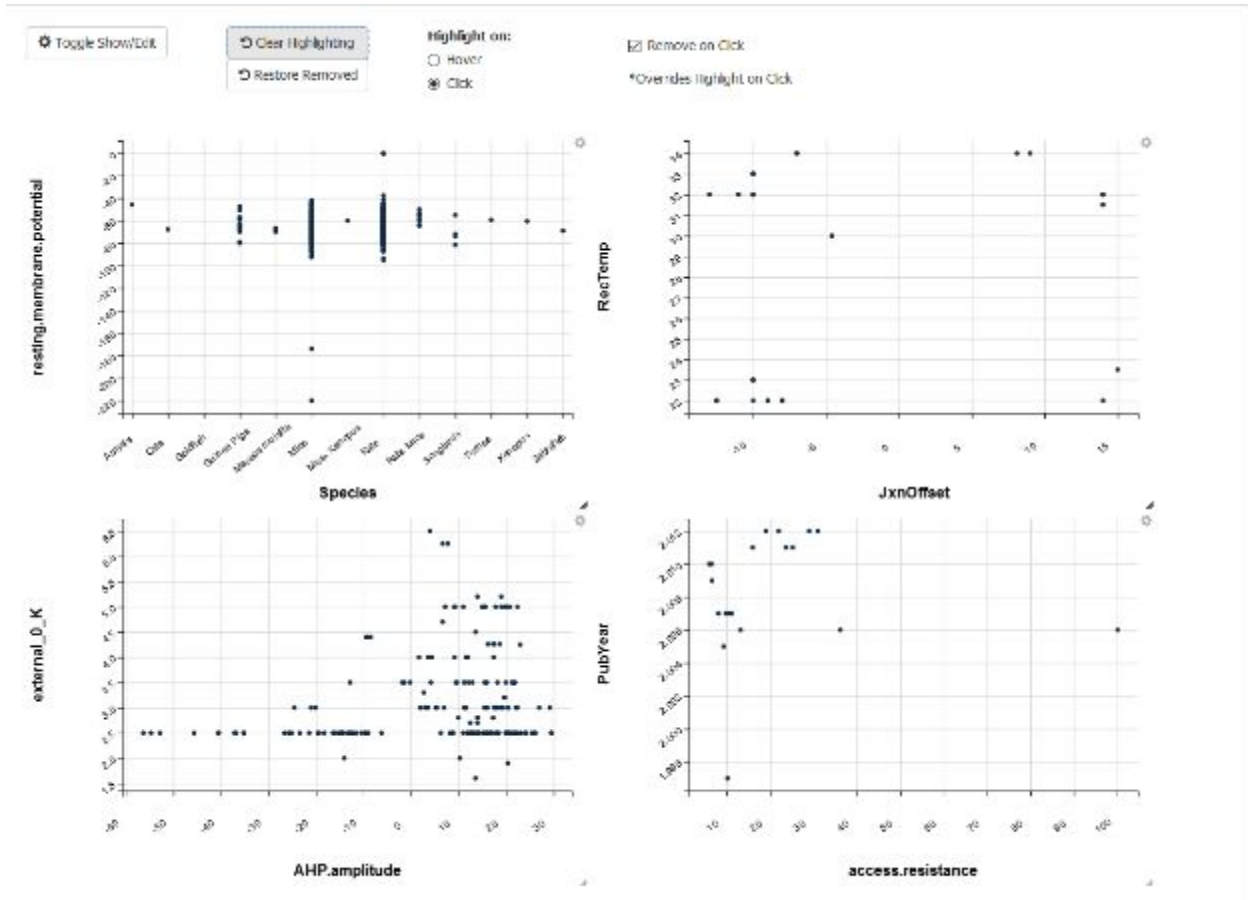


Figure 5. Clear highlighting button pressed, all highlighting removed from plots.



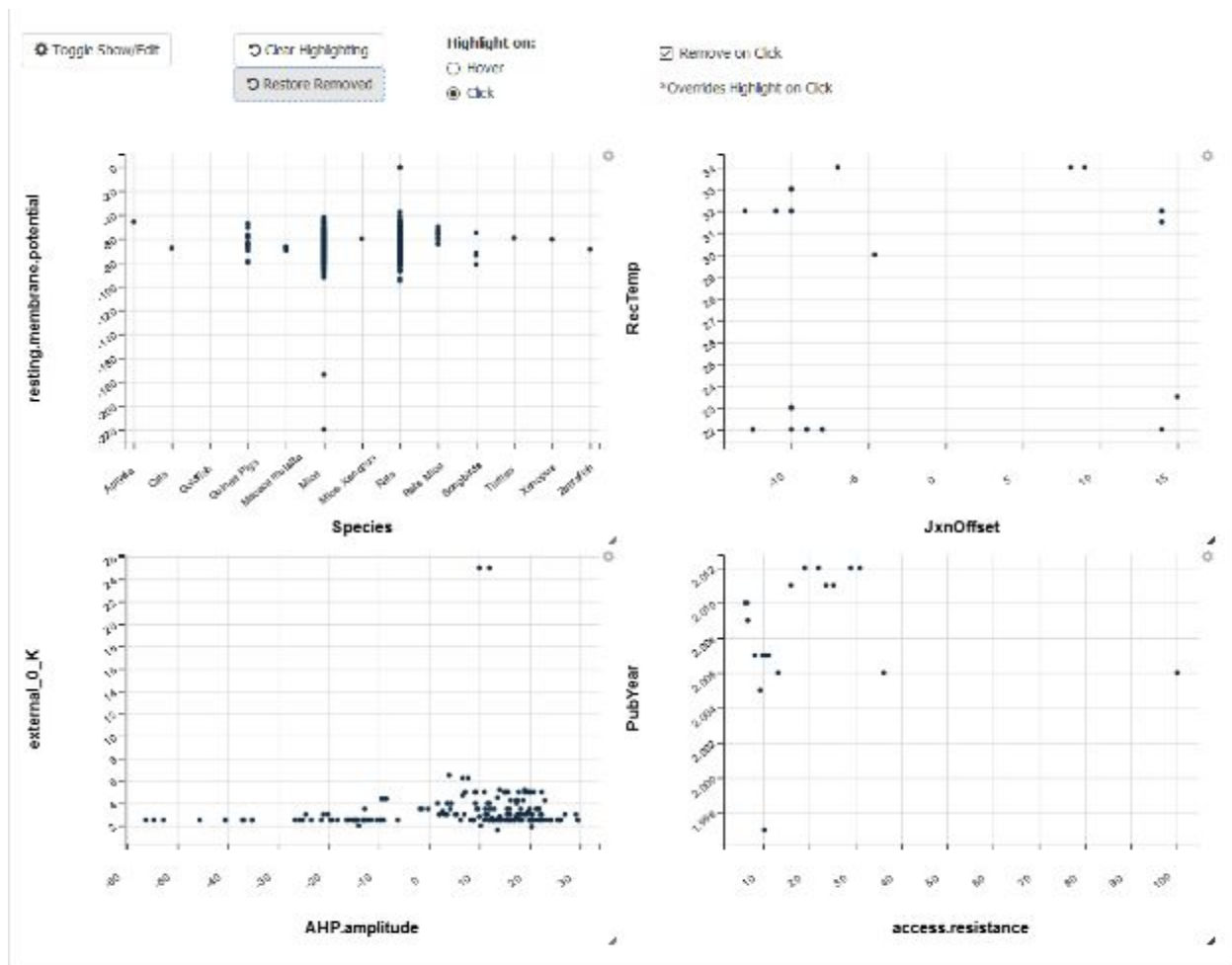


Figure 6. Restore removed button pressed. Removed points are brought back into the plots and they automatically resize.

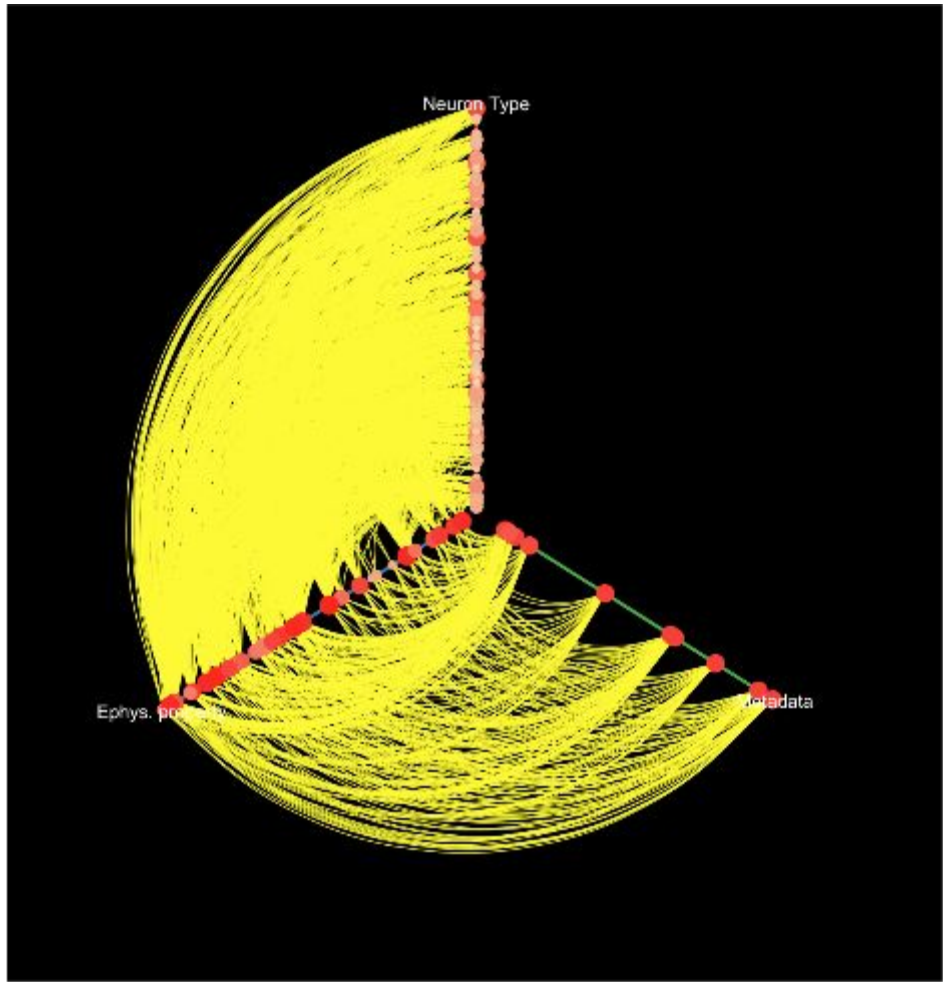


Figure 7. Pac-man. Current version of hive plot figure.