Published in final edited form as:
J Open Source Softw. 2020 ; 5(49): . doi:10.21105/joss.02017.

ggalluvial: Layered Grammar for Alluvial Plots

Jason Cory Brunson[1]

[1]Center for Quantitative Medicine, UConn Health

Summary

Alluvial diagrams use stacked bar plots and variable-width ribbons to represent multi-dimensional or repeated-measures data comprising categorical or ordinal variables (Bojanowski & Edwards, 2016; Rosvall & Bergstrom, 2010). The ggalluvial package extends the layered grammar of graphics of ggplot2 (Wickham, 2016) to generate alluvial diagrams from tidy data (Wickham, 2014).

The package makes two key contributions to the R ecosystem. First, ggalluvial anchors the imprecise notion of an alluvial diagram to the rigid grammar of graphics (Wilkinson, 2006), which lends the plots more precise meaning and opens up many combinatorial possibilities. Second, ggalluvial adopts a distinctive geological nomenclature to distinguish "alluvial plots" and their graphical elements from Sankey diagrams and parallel sets plots, which I hope prove useful as these visualization tools converge toward common standards.

Functionality

The primary vignette thoroughly describes and illustrates the functionality of ggalluvial, and the reader is encouraged to browse the package documentation for comprehensive examples. In brief, the package contains stat and geom functions to add the following layers to a ggplot2 object:

- *strata*, or stacked bar plots, located in parallel along a (plotting) axis of (variable) *axes* or *dimensions*
- *alluvia*, ribbons through strata that connect the categories of individual cases or cohorts at different axes
- *lodes*, subdivisions of strata by their intersections with alluvia
- *flows*, segments of alluvia between strata

Figure 1 illustrates these and other plot elements by visualizing changes in several students' curricula (based on their declared majors) across several academic terms. Each axis corresponds to an odd-valued term (1 through 15), at which the students are grouped into strata according to their curricula—Art History, Ceramic, etc. The individual students can be tracked from term to term along their alluvia: for instance, one student started out in Digital Art, encoded by the blue ribbon, but had switched to Painting by the 11th term, where the ribbon turns pink. The partially transparent flows are colored according to their originating (not their terminating) terms, and the lodes where they intersect the strata are obscured by the solid-colored strata themselves. When a student's curriculum is unknown, they are grouped into the "missing" (`NA`) stratum, which is weighted negatively in this example.

Plot layers are formed by pairing stats (statistical transformations) with geoms (mappings to graphical elements and properties); while every stat and geom has a conventional default, alternative grammatical pairings provide combinatorial richness to plotting possibilities. In the above example, the alluvium geom was paired with the flow stat, so that the flows of each alluvium could change color across the axes. Other meaningful stat–geom combinations can be found in the documentation, including pairings of the three alluvial stats (stratum, alluvium, and flow) with the text, errorbar, and pointrange geoms.

Alluvial layers can interpret tidy data in either of two formats: long (one row per lode) and wide (one row per alluvium). These are related by the pivot operations of tidyr (Wickham et al., 2019) and can be toggled between using the custom functions `to_lodes_form()` and `to_alluvia_form()`. The alluvial stats require custom aesthetics—either `stratum` and/or `alluvium` in combination with `x`, if the data are in long format, or some number of axis specifications (`axis1`, `axis2`, etc.), if the data are in wide format.[1] Because the alluvial geoms are specialized to these stats, no pairings with outside stats are currently supported.

Most of the stat parameters control how the strata at each axis, and the lodes within each stratum, are ordered vertically. By default, these orderings are independent of differentiation aesthetics, so that layers are consistent within and across plots unless otherwise specified. An auxiliary vignette details the effects of these parameters. They can also be set as global options.

Concepts

Visualizations of flow processes have long encoded magnitudes as ribbon widths, constituting a type called Sankey diagrams (Schmidt, 2008). A widely-used subtype for longitudinal categorical data represent categories as nodes threaded by edges that represent the trajectories and magnitudes of cases (Riehmann, Hanfler, & Froehlich, 2005). Their design anticipated parallel sets plots, which were adapted from parallel coordinates plots (Inselberg & Dimsdale, 1987; Wegman, 1990) to visualize multivariate categorical data, and which represent cohorts of equivalent cases as ribbons connecting categories represented as boxes (Kosara, Bendix, & Hauser, 2006). These in turn anticipated "alluvial diagrams", proposed to visualize changes in case memberships across successive cross-sections (Rosvall & Bergstrom, 2010). Several R packages have been developed to generate diagrams of these types, including riverplot (Weiner, 2017), networkD3 (Allaire, Gandrud, Russell, & Yetman, 2017), sankey (Csárdi & Weiner, 2017), alluvial (Bojanowski & Edwards, 2016), ggparallel (Hofmann & Vendettuoli, 2013), ggforce (Pedersen, 2019), ggalluvial (Brunson, 2019), and ggpcp (Ge & Hofmann, 2019).

[1] Because these aesthetics are not recognized by ggplot2, they produce warnings under some conditions.

Sankey, parallel sets, and alluvial diagrams are often conflated, and there is currently no consensus on what features are distinctive to each type. Moreover, their graphical elements go by a variety of names, often interchangeably. In order to more clearly describe the features of ggalluvial in relation to similar packages, I have found it useful to adopt a careful demarcation among these diagram types.

Statistical graphics (here also simply called "plots") are diagrams that communicate statistical information using graphical methods (Friendly, 2005) and, more narrowly, are uniquely determined from data by a fixed set of plotting rules (Wilkinson, 2006). By design, graphics produced by ggplot2 extensions are plots: The stat, geom, and other layers of a ggplot object exactly reproduce a graphic from data (under the same parameter settings).[2] Sankey diagrams are much more flexible. The earliest engine efficiency diagrams in this tradition could take a variety of forms to depict the same energy flow and were differently annotated for different audiences (Schmidt, 2008). Software implementations may use heuristic algorithms to position their graphical elements (Allaire et al., 2017; Csárdi & Weiner, 2017) or enable users to manually, even interactively, adjust them (Allaire et al., 2017; Riehmann et al., 2005; Weiner, 2017). Paradoxically, Sankey diagrams are overwhelmingly used to represent flow, whereas the aforecited ggplot2 extensions are used to visualize a wide variety of data types. Arguably, these extensions are better understood as producing a different type of diagram.

Parallel sets plots might be viewed as a subtype of Sankey diagram with the following features: Ribbons proceed monotonically along one dimension, and every ribbon encounters a box at every axis. These graphical constraints correspond to combinatorial constraints on the data, which amount to an id–key–value structure in which every id–key pair takes exactly one value (possibly zero or missing, and optionally weighted). In this sense, the plots produced by the ggplot2 extensions (and by the alluvial package) are parallel sets plots: Cohorts are partitioned into categories at each axis and connected by ribbons whose widths encode their magnitudes.[3]

The plots produced still vary—in the shapes of ribbons, the arrangements of boxes, and the presence of gaps between boxes at the same axis. The exceptional geoms of ggparallel each offer common-angle as well as linear ribbons. Those of alluvial, ggforce, ggalluvial, and ggpcp offer one-parameter families that interpolate between straight and x-spline ribbons.[4] The stats vertically arrange the elements (boxes and ribbons) at each axis. These distinct elements are rendered by separate layers in ggforce, ggalluvial, and ggpcp, following the additive (+) syntax of ggplot2. ggalluvial provides more levers of control over the statistical transformations, thereby over the messages conveyed by the plot, than the other packages.[5]

[2] There are exceptions. For example, the jitter geom in ggplot2 introduces randomness to symbol positions, and the repel geoms of ggrepel (Slowikowski, 2019) use heuristic algorithms to position text.

[3] The possible exceptions are the hammock plots and common angle plots of ggparallel, which are contrasted with a stricter definition of parallel sets plots than I use here, in which ribbons are straight, their widths aggregate to box widths, and they meet without overlap at the sides of boxes, partitioning them (Hofmann & Vendettuoli, 2013).

[4] Several alternative curves, based on Shaffer (2019), are in development.

[5] Indeed, the dependency package easyalluvial (Koneswarakantha, 2019) was built on top of ggalluvial to exchange much of this flexibility for more expedient data exploration.

The ggalluvial package adopts the term *alluvial plot* for the subtype of parallel sets plots it produces, with the geological terminology introduced above.[6] These alluvial plots are distinguished by two features: a prescribed order on the stacked elements at each axis, including both the values of the discrete variables and the ribbons connecting cases or cohorts between them; and a real-valued plotting dimension perpendicular to that of flow, along which these elements are stacked, so that gaps between them are precluded. In combination, these features confer greater meaning on the second plotting dimension.

The first feature is shared by the other packages but is not essential to parallel sets plots; such plots could, for example, arrange boxes corresponding to repeated categorical decompositions differently at different axes. While most of the packages separate boxes at each axis with gaps, these can be reduced to zero, so that each package can create alluvial plots. (ggparallel and ggalluvial alone *only* produce alluvial plots.) These features are particularly important to some applications and, in my view, can fundamentally change the way a plot is interpreted. It is for this reason that I believe the new typology and terminology are warranted.

Applications

While most uses might be served equally well by other parallel sets plots or Sankey diagrams, alluvial plots seem exceptionally well-suited to three settings: repeated ordinal measures data, incomplete longitudinal data, and signed categorical data.[7]

Repeated ordinal measures data.

Most Sankey, parallel sets, and alluvial implementations stack each bar plot in order of name or of size (though some follow user-provided hierarchies), and most insert gaps between categories for easy visual discrimination. Ordinal variables are most appropriately stacked in their own intrinsic and consistent order and, when the number of categories (hence of gaps) changes from axis to axis, vertical separations can obscure whether magnitude totals changed as well. A use case by Schlotter et al. (2019), to represent patients' physical limitations following an investigational right heart valve repair technique, illustrates the use of an ordinal stratum variable (a heart failure functional classification). Another, by North, Kothe, Klas, & Ling (2019), to represent ranked preferences among several definitions of veganism by survey respondents, illustrates the importance of consistency in their order. In both cases, the fixed heights of the bar plots conveyed that no individuals were lost to follow-up.

Incomplete longitudinal data.

Alluvial plots clearly indicate times at which longitudinal data are censored or otherwise missing: Certain strata, or the alluvia or flows connecting them, are present at one time point but absent at a previous or future one. Seekatz et al. (2018) use this feature to include in one alluvial plot a sample of *Clostridium difficile*–infected patients who had their infections ribotyped at multiple times. Patients were classified by dominant ribotype, and the alluvial

[6] This has the unfortunate side effect of conflating search results from the geology literature.
[7] To be sure, this is a subjective assessment that may be refuted by visualization effectiveness research.

plot showcased variability in this classification. While all 32 patients had at least two samples taken, only 3 had four, communicated by the shortening of the bar plots along the main dimension. Sjoding, Gong, Haas, & Iwashyna (2019) use a similar plot to trace patient groups receiving mechanical ventilation based on discretized tidal volumes, including a grey stratum for patients discontinued from intubation.

Signed categorical data.

Edwards & Pinkerton (2019) produced a novel alluvial plot to represent changes in ownership category of owners in a halibut fishery. The total number of owners changed from year to year as exiters were not exactly matched by new entrants. In order to depict an accurate total but include both new entrants and exiters at each year, the authors affixed a negative stratum for the exiter category to each bar plot.[8] Such a feature has no analogue in Sankey diagrams or parallel sets plots but potentially wide-ranging applications: Bar plots may use “positive” and “negative” bars to represent signed categories, such as contributors to revenue versus deficit, or to contrast the bars divided along a binary variable such as gender across age groups in a population (“pyramid plots”). Alluvial plots provide a way to track cases and cohorts across such graphics, even when cases change sign. Future applications may demonstrate additional uses for this functionality.

Acknowledgments

I am grateful to many users for their feedback on every version of this package. Development benefitted from the use of resources and support of colleagues at UConn Health, and I have been supported in part by T90 training grant 5T90DE021989-07 from the National Institute of Dental and Craniofacial Research.

References

Allaire JJ, Gandrud C, Russell K, & Yetman CJ (2017). networkD3: D3 JavaScript Network Graphs from R. Retrieved from https://cran.r-project.org/package=networkD3

Bojanowski M, & Edwards R (2016). alluvial: R Package for Creating Alluvial Diagrams. Retrieved from https://cran.r-project.org/package=alluvial

Brunson JC (2019). ggalluvial: Alluvial Plots in ‘ggplot2’. Retrieved from https://cran.r-project.org/package=ggalluvial

Csárdi G, & Weiner J (2017). sankey: Illustrate the Flow of Information or Material. Retrieved from https://cran.r-project.org/package=sankey

Edwards DN, & Pinkerton E (2019). Rise of the investor class in the British Columbia Pacific halibut fishery. Marine Policy, 109. doi:10.1016/j.marpol.2019.103676

Friendly M (2005). Milestones in the History of Data Visualization: A Case Study in Statistical Historiography. In Classification — the ubiquitous challenge. Studies in classification, data analysis, and knowledge organization (pp. 34–52). Berlin/Heidelberg: Springer-Verlag. doi:10.1007/3-540-28084-7_4

Ge Y, & Hofmann H (2019). ggpcp: Parallel Coordinate Plots in the ggplot2 Framework. Retrieved from https://yaweige.github.io/ggpcp/

Hofmann H, & Vendettuoli M (2013). Common angle plots as perception-true visualizations of categorical associations. IEEE Transactions on Visualization and Computer Graphics, 19(12), 2297–2305. doi:10.1109/TVCG.2013.140 [PubMed: 24051796]

[8] The authors should be credited with this innovation, which I only implemented in ggalluvial after learning about their workaround to create it using a previous version.

Inselberg A, & Dimsdale B (1987). Parallel Coordinates for Visualizing Multi-Dimensional Geometry. In Computer graphics 1987 (pp. 25–44). Tokyo: Springer Japan. doi:10.1007/978-4-431-68057-4_3

Koneswarakantha B (2019). easyalluvial: Generate Alluvial Plots with a Single Line of Code. Retrieved from https://cran.r-project.org/package=easyalluvial

Kosara R, Bendix F, & Hauser H (2006). Parallel sets: Interactive exploration and visual analysis of categorical data. In IEEE transactions on visualization and computer graphics (Vol. 12, pp. 558–568). doi:10.1109/TVCG.2006.76 [PubMed: 16805264]

North M, Kothe E, Klas A, & Ling M (2019). “It’s not just a diet, it’s a lifestyle”: An Exploratory Study into Community Preferences of Vegan Definitions. PsyArXiv. doi:10.31234/OSF.IO/MKQN4

Pedersen TL (2019). ggforce: Accelerating ‘ggplot2’. Retrieved from https://cran.r-project.org/package=ggforce

Riehmann P, Hanfler M, & Froehlich B (2005). Interactive Sankey diagrams. In IEEE symposium on information visualization, 2005. INFOVIS 2005. (pp. 233–240). IEEE. doi:10.1109/INFVIS.2005.1532152

Rosvall M, & Bergstrom CT (2010). Mapping Change in Large Networks. (Rapallo F,Ed.)PLoS ONE, 5(1), e8694. doi:10.1371/journal.pone.0008694 [PubMed: 20111700]

Schlotter F, Orban M, Rommel K, Besler C, Roeder M, Braun D, Unterhuber M, et al. (2019). Aetiology-based clinical scenarios predict outcomes of transcatheter edge-to-edge tricuspid valve repair of functional tricuspid regurgitation. European Journal of Heart Failure, 21(9), 1117–1125. doi:10.1002/ejhf.1547 [PubMed: 31359620]

Schmidt M (2008). The Sankey Diagram in Energy and Material Flow Management. Journal of Industrial Ecology, 12(1), 82–94. doi:10.1111/j.1530-9290.2008.00004.x

Seekatz AM, Wolfrum E, DeWald CM, Putler RKB, Vendrov KC, Rao K, & Young VB (2018). Presence of multiple Clostridium difficile strains at primary infection is associated with development of recurrent disease. Anaerobe, 53, 74–81. doi:10.1016/j.anaerobe.2018.05.017 [PubMed: 29859301]

Shaffer J (2019, May 13). Sankey diagrams: Why i used the sigmoid function and why you probably shouldn’t. https://www.dataplusscience.com/Sigmoid.html.

Sjoding MW, Gong MN, Haas CF, & Iwashyna TJ (2019). Evaluating Delivery of Low Tidal Volume Ventilation in Six ICUs Using Electronic Health Record Data. Critical Care Medicine, 47(1), 56–61. doi:10.1097/CCM.0000000000003469 [PubMed: 30308549]

Slowikowski K (2019). ggrepel: Automatically Position Non-Overlapping Text Labels with ‘ggplot2’.

Wegman EJ (1990). Hyperdimensional Data Analysis Using Parallel Coordinates. Journal of the American Statistical Association, 85(411), 664–675. doi:10.1080/01621459.1990.10474926

Weiner J (2017). riverplot: Sankey or Ribbon Plots. Retrieved from https://cran.r-project.org/package=riverplot

Wickham H (2014). Tidy Data. Journal of Statistical Software, 59(10), 1–23. doi:10.18637/jss.v059.i10 [PubMed: 26917999]

Wickham H (2016). ggplot2: Elegant Graphics for Data Analysis. New York: Springer-Verlag. doi:10.1007/978-0-387-98141-3

Wickham H, Averick M, Bryan J, Chang W, D’Agostino LM, François R, Grolemund G, et al. (2019). Welcome to the Tidyverse. Journal of Open Source Software, 4(43), 1686. doi:10.21105/joss.01686

Wilkinson L (2006). The Grammar of Graphics (2nd ed.). Springer Science & Business Media. doi:10.1007/0-387-28695-0

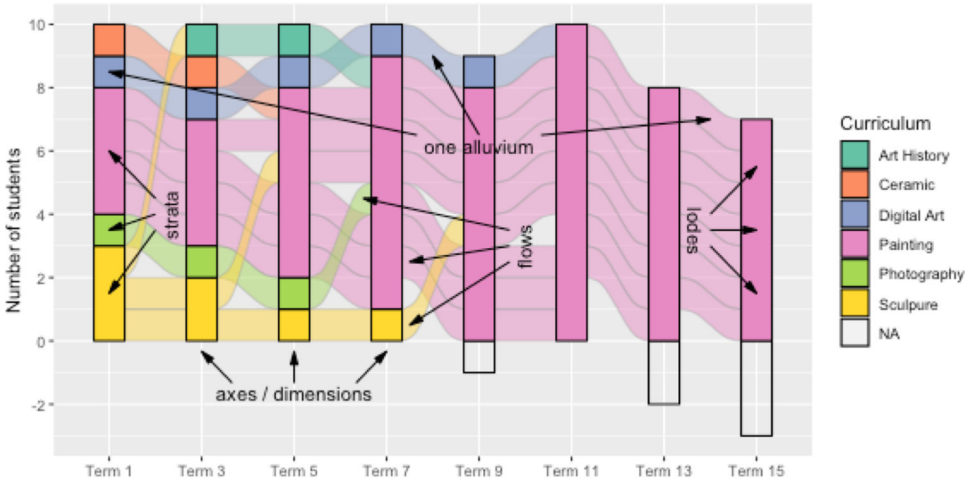

Figure 1:

Alluvial plot of changes in curricula by a cohort of art students