Bioinformatics, 2018, 1–3
doi: 10.1093/bioinformatics/bty109
Advance Access Publication Date: 23 February 2018
Applications Note

OXFORD

Data and text mining

nVenn: generalized, quasi-proportional Venn and Euler diagrams

José G. Pérez-Silva, Miguel Araujo-Voces and Víctor Quesada*

Departamento de Bioquímica y Biología Molecular, Universidad de Oviedo, Oviedo 33006, Spain

*To whom correspondence should be addressed.
Associate Editor: Jonathan Wren

Received on December 7, 2017; revised on February 20, 2018; editorial decision on February 21, 2018; accepted on February 21, 2018

Abstract

Motivation: Venn and Euler diagrams are extensively used for the visualization of relationships between experiments and datasets. However, representing more than three datasets while keeping the proportions of each region is still not feasible with existing tools.
Results: We present an algorithm to render all the regions of a generalized n-dimensional Venn diagram, while keeping the area of each region approximately proportional to the number of elements included. In addition, missing regions in Euler diagrams lead to simplified representations. The algorithm generates an n-dimensional Venn diagram and inserts circles of given areas in each region. Then, the diagram is rearranged with a dynamic, self-correcting simulation in which each set border is contracted until it contacts the circles inside. This algorithm is implemented in a C++ tool (nVenn) with or without a web interface. The web interface also provides the ability to analyze the regions of the diagram.
Availability and implementation: The source code and pre-compiled binaries of nVenn are available at https://github.com/vqf/nVenn. A web interface for up to six sets can be accessed at http://degradome.uniovi.es/cgi-bin/nVenn/nvenn.cgi.
Contact: quesadavictor@uniovi.es
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

A recurrent task in data mining is set visualization (Alsallakh *et al.*, 2016). Ideally, the aim of this analysis is to find the most important relationships between datasets at a glance (Lex *et al.*, 2014). Venn and Euler diagrams are a popular tool for this purpose, as they represent in a single figure all the relevant overlaps between sets. Venn diagrams are similar to Euler representations, but they show all the possible intersections between sets, even if they do not exist in the input. In the field of bioinformatics, sets can for instance contain genes that are differentially expressed in multiple conditions. Similarities in the response to those conditions will be immediate apparent as intersections containing a larger-than-expected number of elements. For this reason, making the area of each region proportional to the number of elements it contains is particularly useful.

Multiple tools exist for the automatic creation of Euler diagrams, reflecting the extensive use of this representation in research. Most of these tools represent up to three sets, and keep the regions approximately proportional to the number of elements (e. g., Hulsen *et al.*, 2008, Micallef and Rodgers, 2014a). Representing more than three sets while keeping proportionality is not trivial, as symmetric set shapes are not flexible enough. Several tools accomplish approximate proportionality by using penalty functions or other transformations (e. g., Kestler *et al.*, 2008). However, most tools simply present a pre-drawn n-set Venn diagram with numbers inserted, which, while useful, is hard to interpret (e. g., Bardou *et al.*, 2014, Heberle *et al.*, 2015).

An intriguing development in the creation of proportional Euler diagrams has been used in eulerForce (Micallef and Rodgers, 2014b). The algorithm used inside this tool performs a physical simulation on a system which is attuned to generate the desired layout for an Euler diagram. Each curve enclosing a set is represented conceptually as a number of charges joined by springs. By manipulating the forces between those virtual charges, eulerForce creates Euler diagrams that are regular, smooth and aesthetically pleasing.

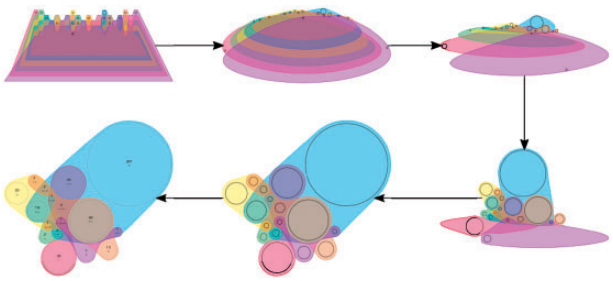

Fig. 1. nVenn algorithm. The figure is created through simulation from a generalized Venn diagram (top left) up to a quasi-static solution (bottom center). Then, the lines are softened to obtain the final figure (bottom left)

However, the areas of the resulting regions are not easy to manipulate, and using more than five curves is too computationally expensive. These limitations illustrate the complexity of representing multiple sets with symmetric or simple convex shapes and respect the proportionality of each region.

Here, we present an algorithm to create Venn and Euler diagrams with an arbitrary number of sets. Each region in the diagram contains a circle whose area is proportional to the number of elements in that intersection. We have also prepared a web interface to create proportional diagrams with up to six sets.

2 Materials and methods

The nVenn program, coded in C++, accepts a text input describing the sizes of each region in a Venn diagram and outputs a figure in SVG format. The steps of this algorithm are summarized in Figure 1. Briefly,

- A generalized Venn diagram for the desired number of sets (n) is generated based on a symmetric chain decomposition (Greene and Kleitman, 1976; Griggs *et al.*, 2004). First, the boolean lattice for n groups is generated based on the depiction in Ruskey *et al.* (2006). This lattice expresses each region as a boolean vector where each element represents a set. If that position is filled with a 1, the region belongs to the set, whereas if it is filled with a 0 it does not belong to the set. The symmetric chain decomposition ensures that all the regions belonging to a set can be enclosed with a simple curve while excluding all the regions not belonging to the set (Supplementary Material, Section 1.1).
- Then, each region is shrunk through simulation (Fig. 1, top center, top right, bottom right and bottom center). In this process, each line of the diagram is replaced by a large number of points joined by springs (Supplementary Material, Section 1.2). Internal circles move when contacted by line points and line springs. This system is simulated with a naïve engine based on a small delta time. It includes friction and damping forces to speed up the contraction of lines.
- Finally, the contracted diagram is embellished through a simulation where inner circles are fixed and an attractive spring force (Supplementary Material, Section 1.1) between line points and circles is added, so that lines represent each region more closely (Supplementary Movie S1, starting at 20 seconds).

The end of each step is controlled by the user in different ways, depending on the interface.

2.1 Interfaces

The current version of nVenn can be used with three different interfaces: command line, OpenGL graphical output and web interface. Although the core methods are the same, each flavor has distinct requirements and modes of use.

2.1.1 Command line

This version accepts a text input file describing each region in the diagram (Supplementary Fig. S1) and performs all three steps automatically. The number of cycles per step is fixed in the code, and fits most purposes for up to six sets. In the current version, the main step consists of 7000 cycles. The program automatically saves the intermediate result, so that more cycles can be added by simply repeating this procedure on the same input file. The final embellishment runs for 200 cycles. The syntax to run this version is:

```
./nVenn input_file [output_file_name=result]
```

By repeated execution, an unlimited number of sets can be processed. However, it must be noted that the time it takes for the simulation to complete grows quickly with the number of sets. This version uses standard libraries, and therefore it is easy to compile in most operative systems. x64 Linux Debian and Microsoft Windows pre-compiled versions are available for download.

2.1.2 Graphical output

A graphical version using OpenGL is also provided. The input for this tool has the same format as that of the command-line version, although the names of input and output files are fixed. By contrast, the user can decide in real time when to jump from one phase to the next. This interface also gives a simple overview of the simulation process, as shown in Supplementary Movie S1.

The OpenGL interface uses Microsoft Windows-specific libraries. A pre-compiled version for this platform is available for download.

2.1.3 Web interface

This version uses a more simple input and allows further analysis of the output diagram. Thus, users can directly enter the members of each set in text boxes. The interface then calculates the number of elements in each region of the diagram, runs nVenn and renders the output. In addition, users can query their data for any intersection between sets by checking boxes or by directly clicking the corresponding region in the output figure (Fig. 2). The length of the simulation is fixed, but more cycles can be added by repeated execution. At this time, the interface allows up to six sets, but the system is easily scalable to a higher number of sets.

The interface also allows the customization of the final figure. Users can add or remove labels to describe each region and to show how many elements are included in it. The color and opacity of each set, as well as the width of their borders, can also be tweaked. The output figure can be saved in a vectorial format (scalable vector graphics, SVG) and in a bitmap format (portable network graphics, PNG).

The web page interfacing nVenn is coded in standard HTML, CSS (using min. Bootstrap v.3.3.1) and javascript (using min. jQuery v.1.11.1) We have also added a step-by-step tutorial at the top navigation bar.

2.2 Test

As a proof of concept, we performed an Euler diagram with the web interface of nVenn using genes included in the *innate immune system* GO category (GO: 0045087). Six subsets with different GO evidence codes were generated: IBA (Inferred from Biological aspect of Ancestor), IC (Inferred by Curator), IDA (Inferred from Direct

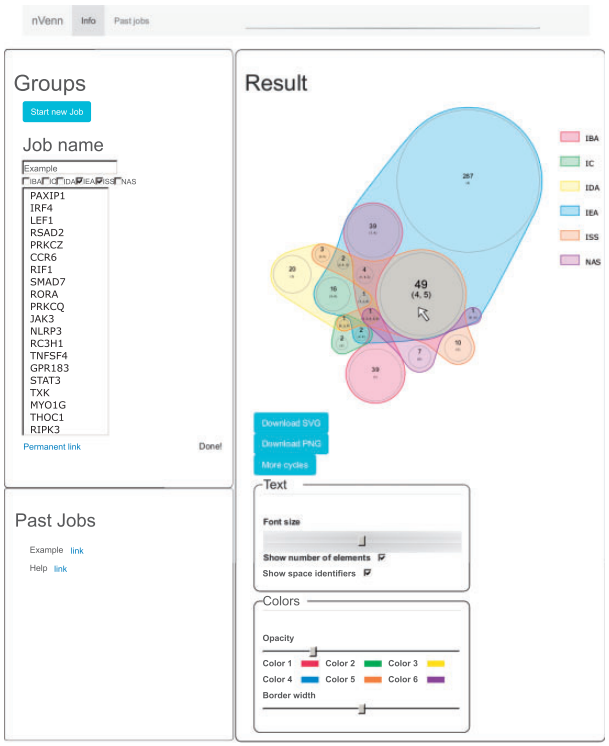

Fig. 2. Test nVenn diagram in web interface. The figure is shown at the right panel, with the tools provided for manipulation below. The text area at the top left panel shows the elements present only in sets 4 and 5. This region can be selected from the checkboxes (above the text area) and by clicking the corresponding area in the figure (arrow)

Assay), IEA (Inferred from Electronic Annotation), ISS (Inferred from Sequence or Structural Similarity) and NAS (Non-traceable Author Statement). The execution took three submissions for a satisfying result (about 12 min).

The result is shown in Figure 2, and gives a quick overview of the different sizes of each subset, as well as conspicuous correlations between them. Thus, the intersection of the IEA and ISS groups is so large as to make ISS almost a subset of IEA. Also, the number of elements shared only by those subsets is disproportionately large. Figure 2 also shows how users can analyze each region by just clicking on it.

3 Discussion

The aim of nVenn is to produce easy-to-interpret Euler diagrams that convey information about an unlimited number of sets. Although this capability exists, in practice it is very hard to interpret Venn diagrams for more than six sets.

The simulation-based algorithm for set drawing is conceptually similar to that of eulerForce, although the latter only simulates lines. In fact, the aim of eulerForce is building well-formed Euler diagrams. Achieving proportionality with this program would be very hard, requiring users to provide the coordinates of a starting Euler diagram and to perfectly balance the internal forces. By contrast, the use of inner circles in nVenn allows users to directly control the size of each region. This means that the resulting diagrams are not necessarily convex or strictly well-formed as with eulerForce. However, we have added specific steps to ease the interpretation of the results.

Thus, in the development of nVenn, some aesthetic qualities of the final diagram have taken precedence over strict proportionality. First, the area of each region is larger than the inner proportional circle, which produces smoother curves for each set. Therefore, users must take into account the areas of circles, and not empty spaces. In this regard, large numbers of sets with few intersections will frequently lead to diagrams with large empty spaces (Supplementary Fig. S2). Since this algorithm tackles a hard circle-packing problem with added constraints, this drawback is expected and accepted. Future versions of nVenn may incorporate random deviations in the initial conditions so that, after multiple runs, several solutions can be explored.

Further deviations from proportionality occur when some of the regions are too small in the final figure. To avoid invisible regions, the minimal circle radius is set at 1% of the width or height of the diagram. Circles whose radius should be lower than that will appear larger than expected. Since those regions would be even harder to interpret in a strict diagram, this caveat is also accepted. Finally, the set lines are separated by different distances from the inner circles. This feature minimizes the overlaps between lines so that each set line can be easily followed.

The web interface to nVenn offers additional tools for the analysis of diagrams, so that users can quickly find out which elements correspond to each region. We have designed this interface to be easy and intuitive, in the hope that it may become a valuable tool for researchers trying to visualize complex relationships between sets.

Acknowledgements

We thank Drs. Carlos López-Otín, Gloria Velasco and Magda R. Hamczyk for helpful discussions during the development of this manuscript.

Funding

This work has been supported by the Ministerio de Economía y Competitividad-Spain (SAF2014-59986-R, including FEDER funding, and Ramón y Cajal program), Instituto de Salud Carlos III and Principado de Asturias, including FEDER funding.

Conflict of Interest: none declared.

References

Alsallakh,B. *et al.* (2016) The state-of-the-art of set visualization. *Comput. Graph. Forum*, **35**, 234–260.

Bardou,P. *et al.* (2014) jvenn: an interactive venn diagram viewer. *BMC Bioinformatics*, **15**, 293.

Greene,C. and Kleitman,D.J. (1976) Strong versions of sperner's theorem. *J. Combin. Theory Ser. A*, **20**, 80–88.

Griggs,J. *et al.* (2004) Venn diagrams and symmetric chain decompositions in the boolean lattice. *Electronic J. Combin.*, **11**. Research Paper #R2.

Heberle,H. *et al.* (2015) Interactivenn: a web-based tool for the analysis of sets through venn diagrams. *BMC Bioinformatics*, **16**, 169.

Hulsen,T. *et al.* (2008) Biovenn – a web application for the comparison and visualization of biological lists using area-proportional venn diagrams. *BMC Genomics*, **9**, 488.

Kestler,H.A. *et al.* (2008) Vennmaster: area-proportional euler diagrams for functional go analysis of microarrays. *BMC Bioinformatics*, **9**, 67.

Lex,A. *et al.* (2014) Upset: visualization of intersecting sets. *IEEE Trans. Vis. Comput. Graph.*, **20**, 1983–1992.

Micallef,L. and Rodgers,P. (2014a) eulerape: drawing area-proportional 3-venn diagrams using ellipses. *PLoS One*, **9**, e101717.

Micallef,L. and Rodgers,P. (2014b) eulerForce: force-directed layout for euler diagrams. *J. Vis. Lang. Comput.*, **25**, 924–934.

Ruskey,F. *et al.* (2006) The search for simple symmetric venn diagrams. *Notices Am. Math. Soc.*, **53**, 1304–1312.