OntarioTech
Engineering
& Applied Science

A Tour Through

The Visualization Zoo

Presented By:

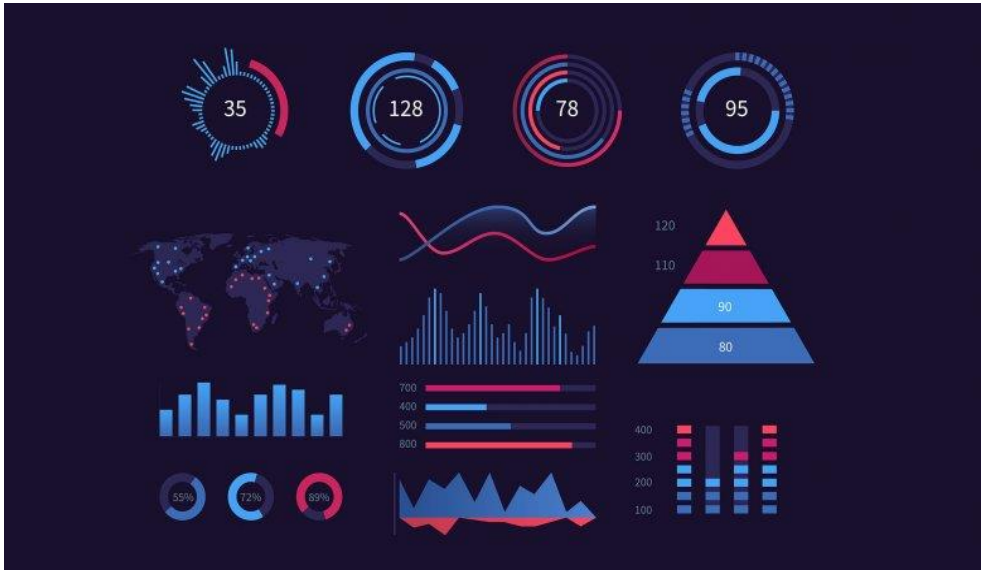
- Dipkumar Patel
- Noopa Jagadeesh
- Prasanth Varma

Topics

1. Introduction
2. Time Series Data Visualization
3. Statistical Distributions
4. Maps
5. Hierarchies
6. Networks
7. Conclusion

Introduction

- What is Data Visualization and why is it important ?
- The challenge is to create effective and engaging visualizations that are appropriate to the data.

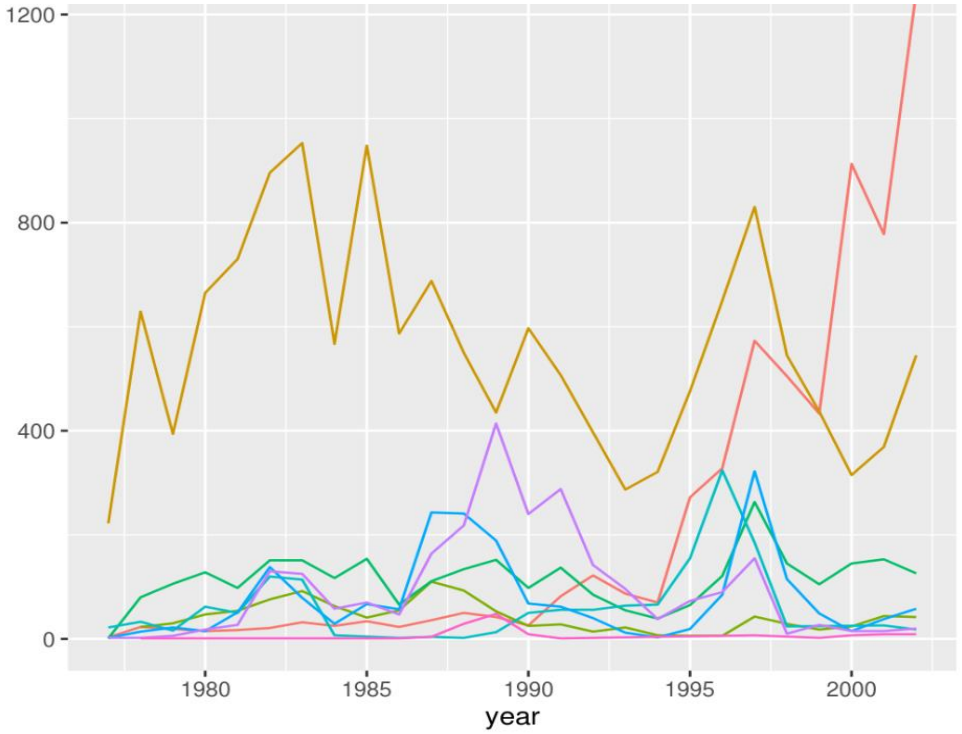


Introduction (cont.)

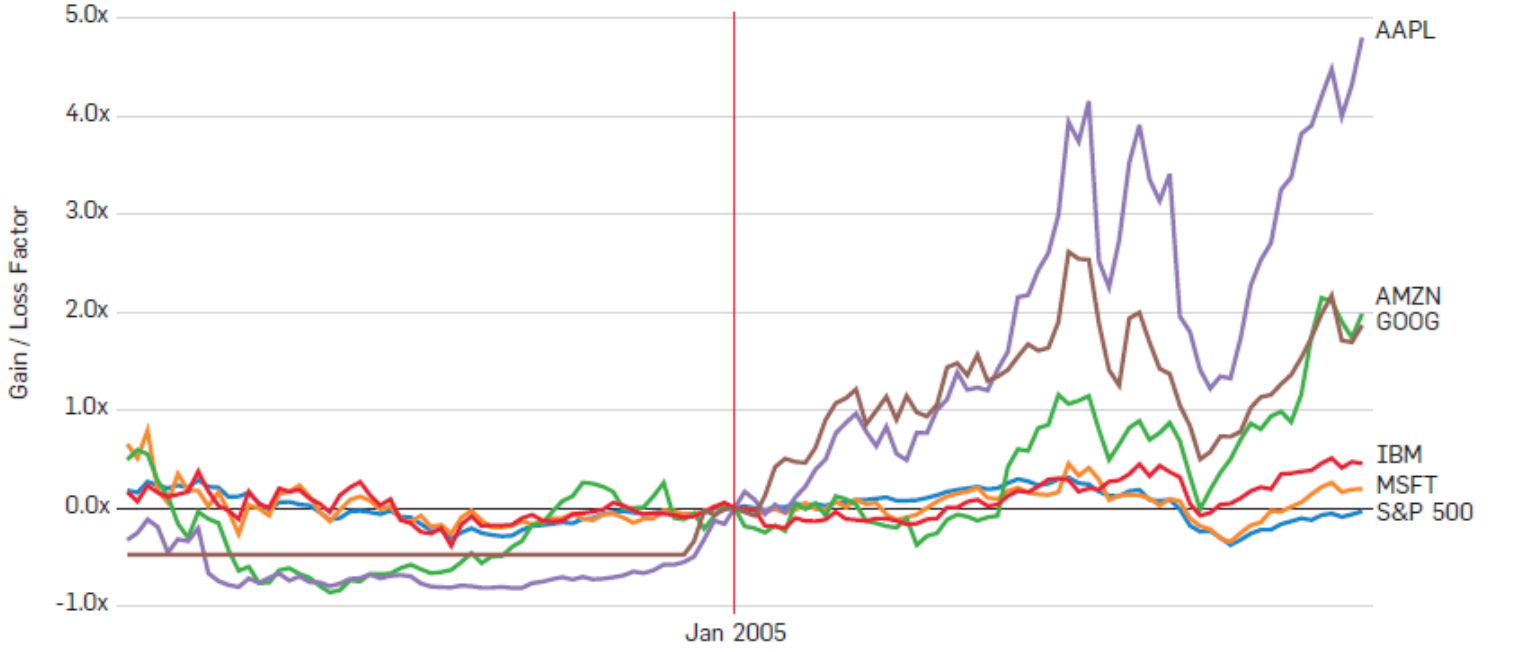
- For creating an effective visualization one must determine which questions to ask, identify the appropriate data, and select effective visual encodings to map data values.
- By now, you might be wondering what is Visualization Zoo?

Time-Series Data Visualization

1. Index Charts
2. Stacked Graphs
3. Small Multiples
4. Horizon Graphs

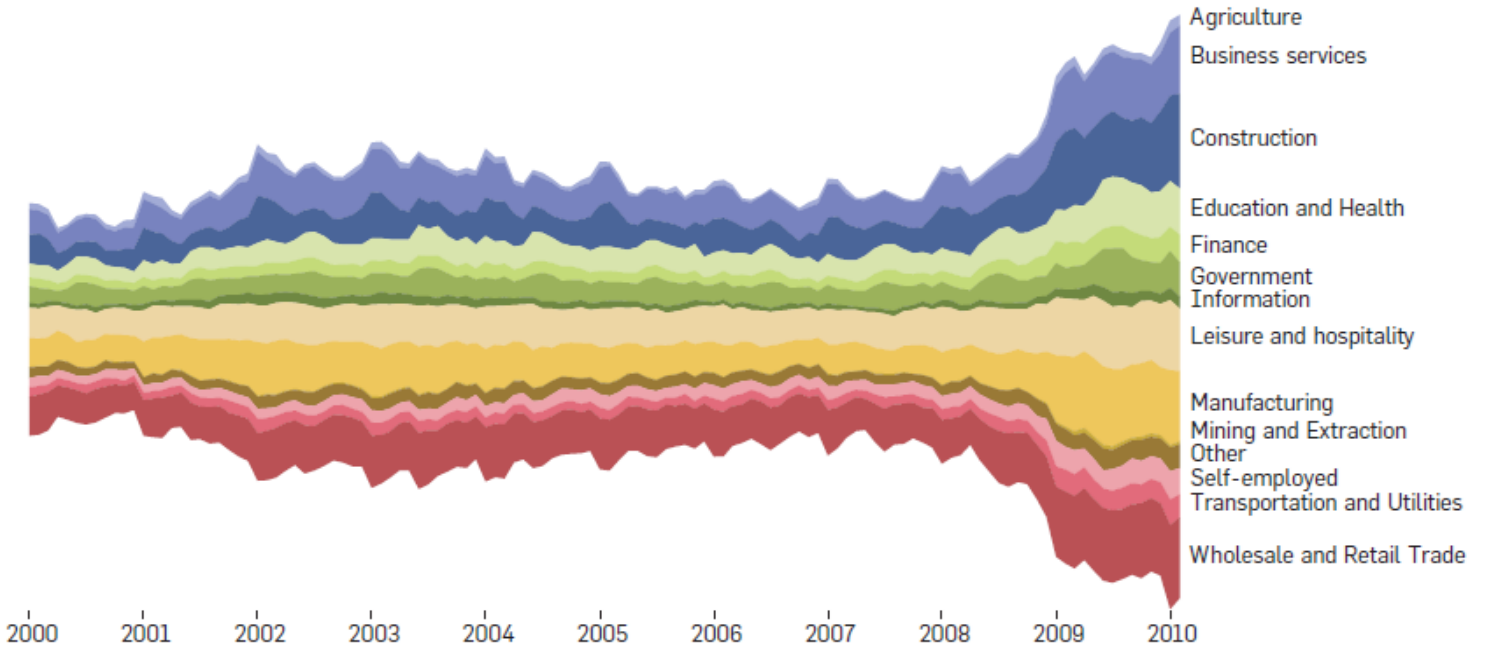


1. Index Charts



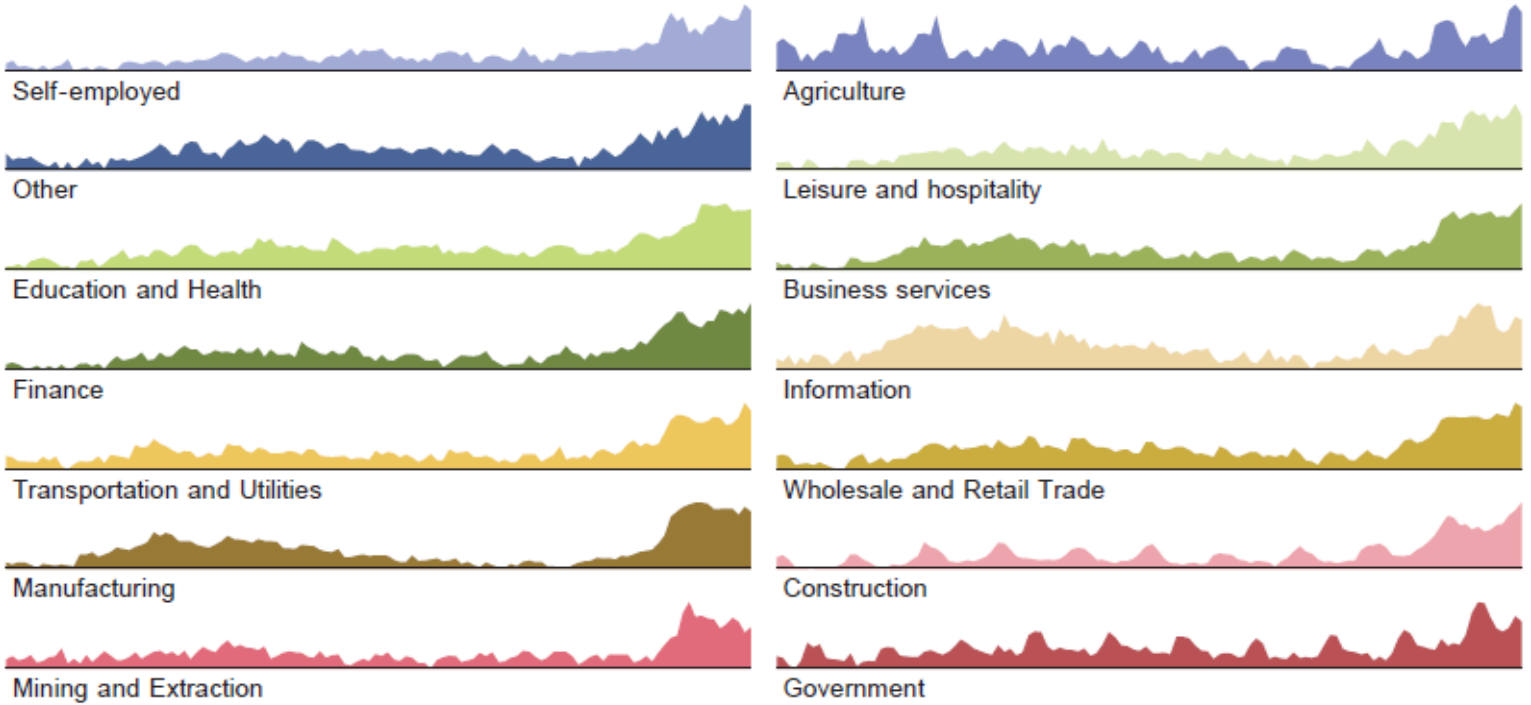
Source: Yahoo! Finance; http://hci.stanford.edu/jheer/files/zoo/ex/time/index-chart.html

2. Stacked Graphs



Source: U.S. Bureau of Labor Statistics; http://hci.stanford.edu/jheer/files/zoo/ex/time/stack.html

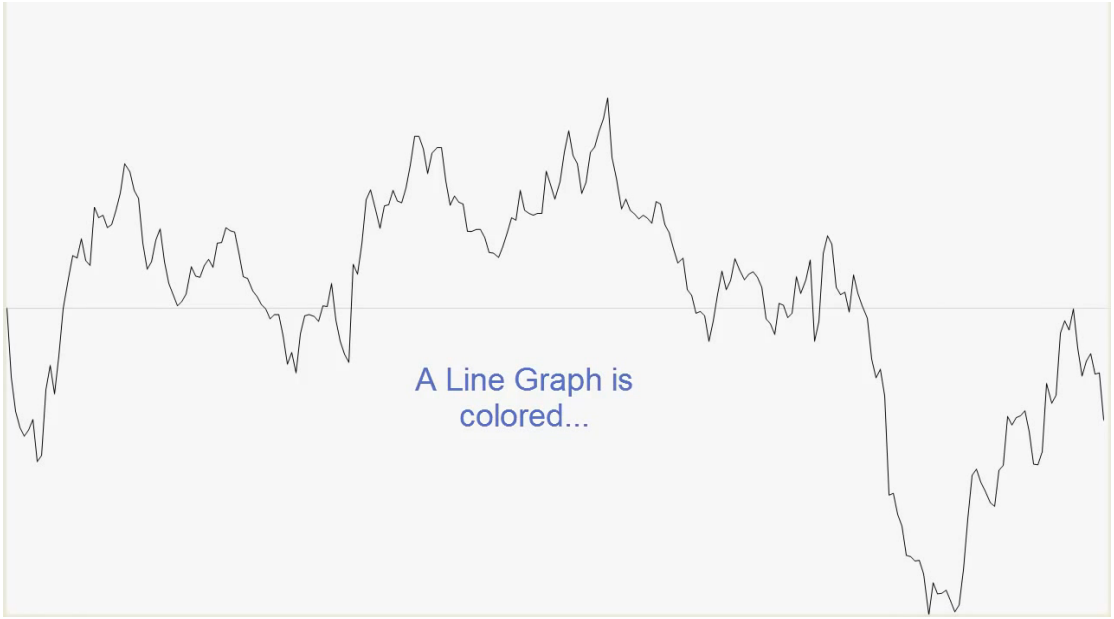
3.Small Multiples



Source: U.S. Bureau of Labor Statistics; http://hci.stanford.edu/jheer/files/zoo/ex/time/multiples.html

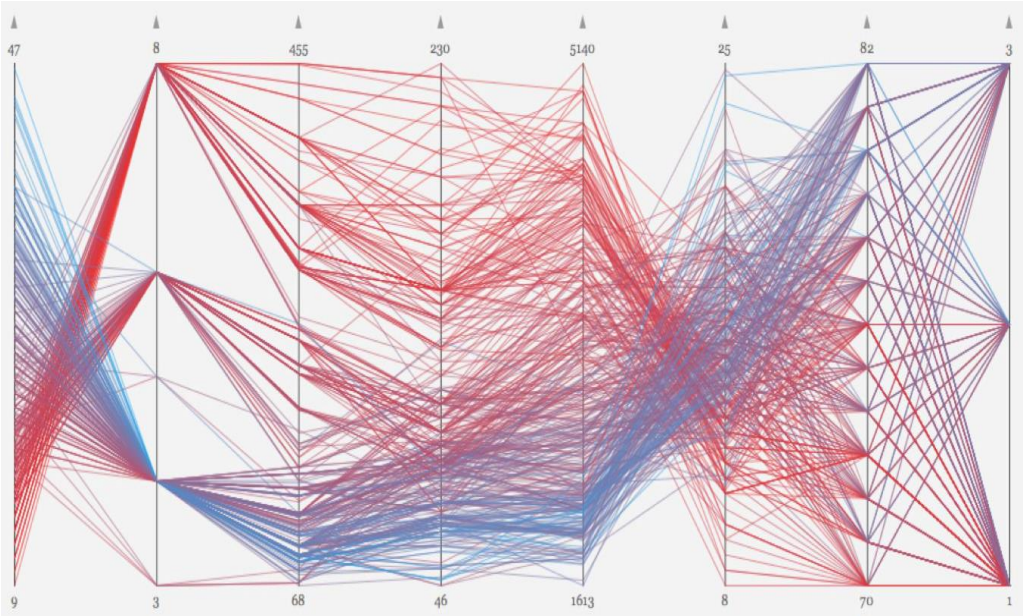
4. Horizon Graphs

- The horizon graph is a technique for increasing the data density of a time-series view while preserving resolution.



Statistical Distributions Visualization

1. Stem-and-Leaf Plots
2. Q-Q Plots
3. SPLOM (Scatter Plot Matrix)
4. Parallel Coordinates



1. Stem-and-Leaf Plots

- A stem-and-leaf plot is a device for presenting quantitative data in a graphical format, similar to a histogram.
- The main advantage of this plot over histogram is that it can be easily built and is convenient to use in determining median or mode of a data set quickly.
- Sorting is performed before the plotting the numbers.

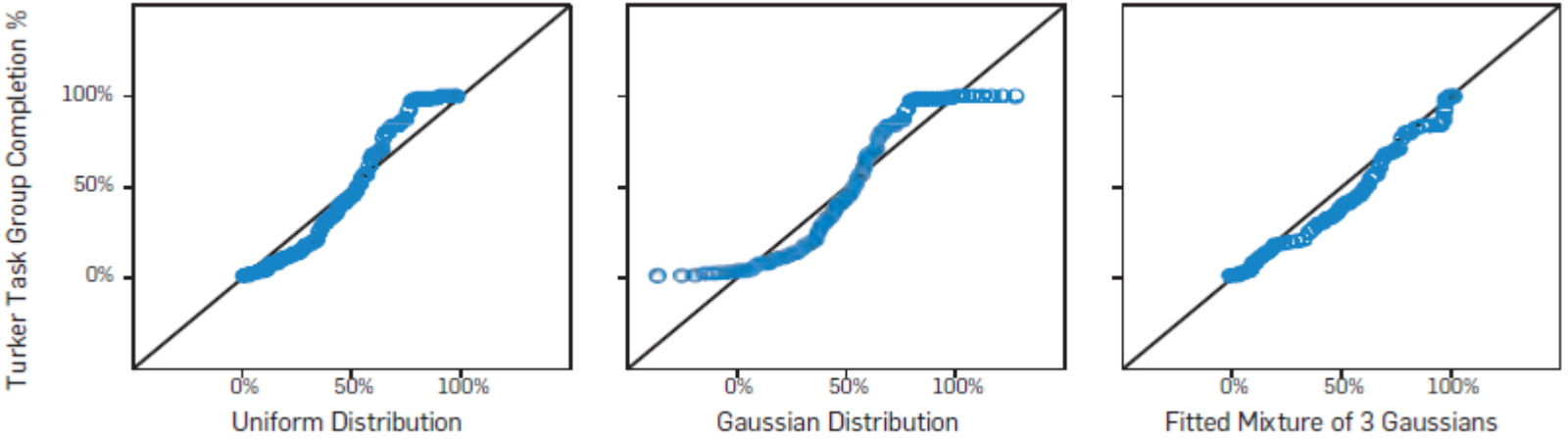
2.3, 2.5, 2.5, 2.7, 2.8 3.2, 3.6, 3.6, 4.5, 5.0

And here is the stem-and-leaf plot:

Stem	Leaf
2	3 5 5 7 8
3	2 6 6
4	5
5	0

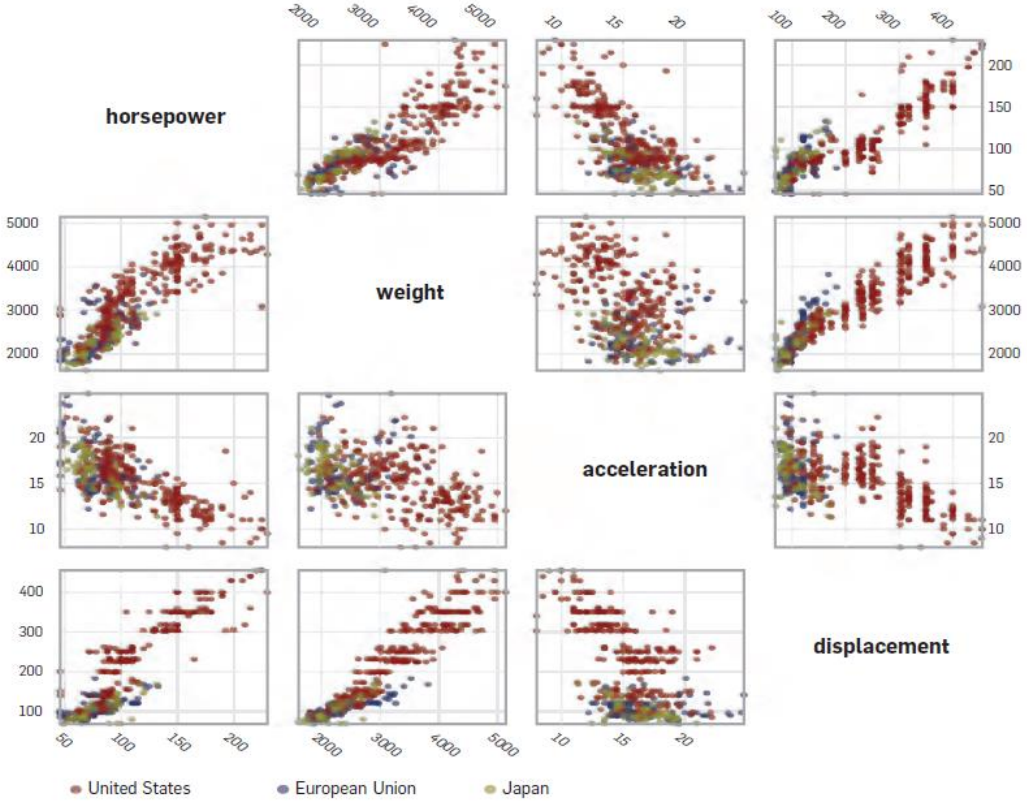
Stem "2" Leaf "3" means **2.3**

2. Q-Q Plots



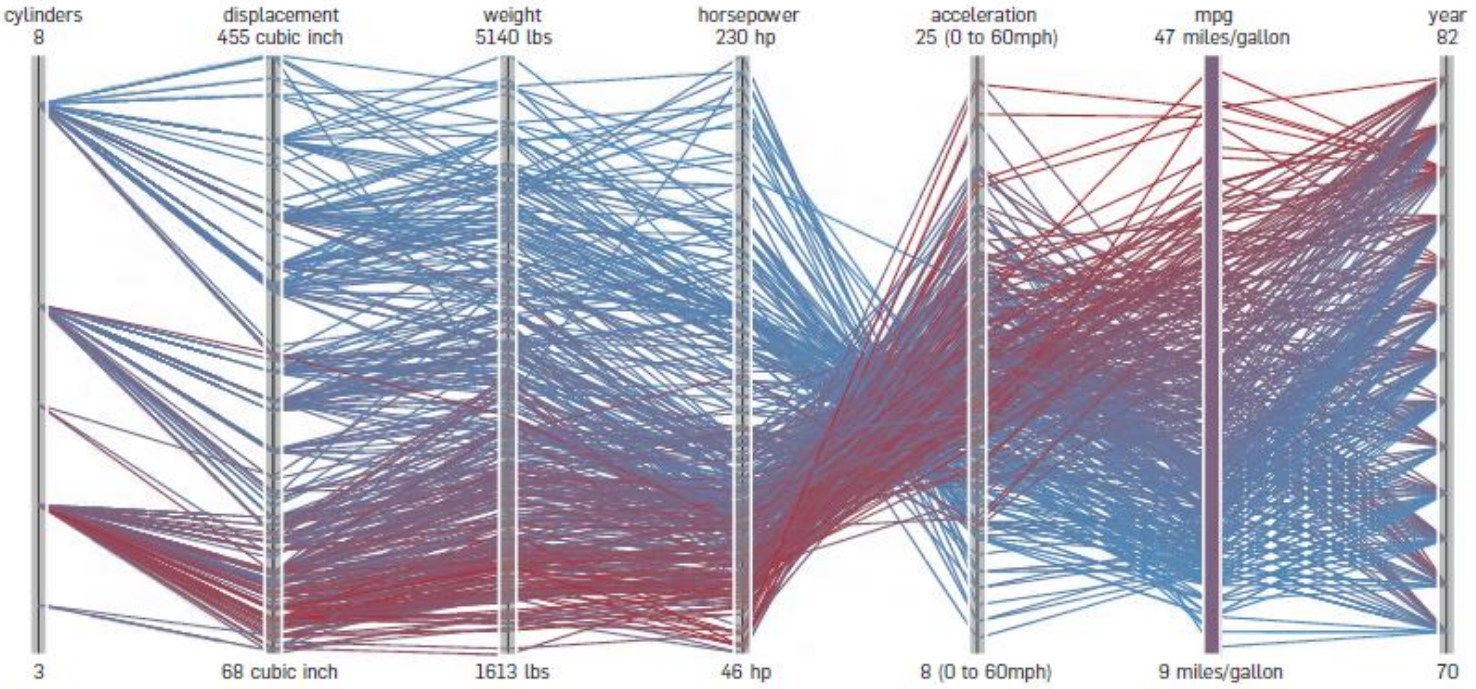
Source: Stanford Visualization Group; http://hci.stanford.edu/jheer/files/zoo/ex/stats/qqplot.html

3. SPLOM

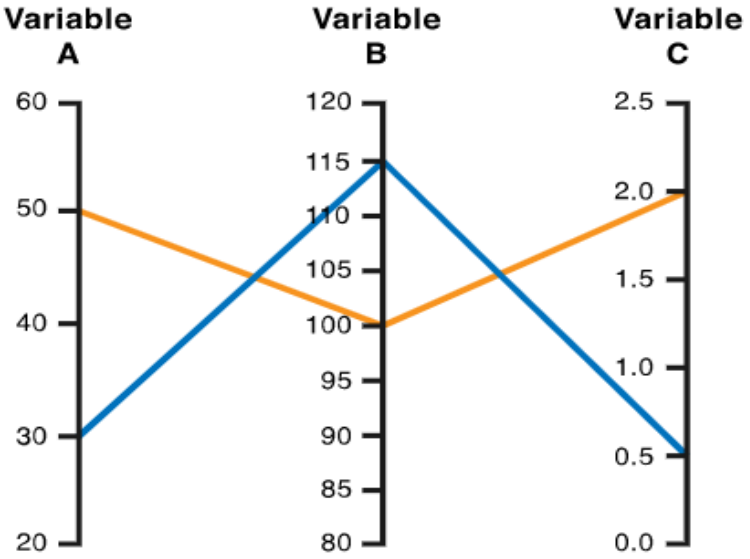


Source: GGobi; http://hci.stanford.edu/jheer/files/zoo/ex/stats/splom.html

4. Parallel Coordinates



Source: GGobi; http://hci.stanford.edu/jheer/files/zoo/ex/stats/parallel.html



Data			
	Variable A	Variable B	Variable C
Item 1	50	100	2.0
Item 2	30	115	0.5

Maps Visualization

Cart-graphic projection: A mathematical function which maps the 3D geometry of earth to a 2D image.

These visualization uses distort or abstract geographic features to highlight the data.

1. Flow Maps

- Flow maps are a combination of **maps** and **flowchart** which indicates the movement of objects from one location to another.
- Lines are used to symbolize the flow. width represents differences in the quantity of the flow.
- Flow lines encode a large amount of multivariate information.
  - Path points
  - Direction
  - Line thickness
  - Color - Present dimension of information to the viewer

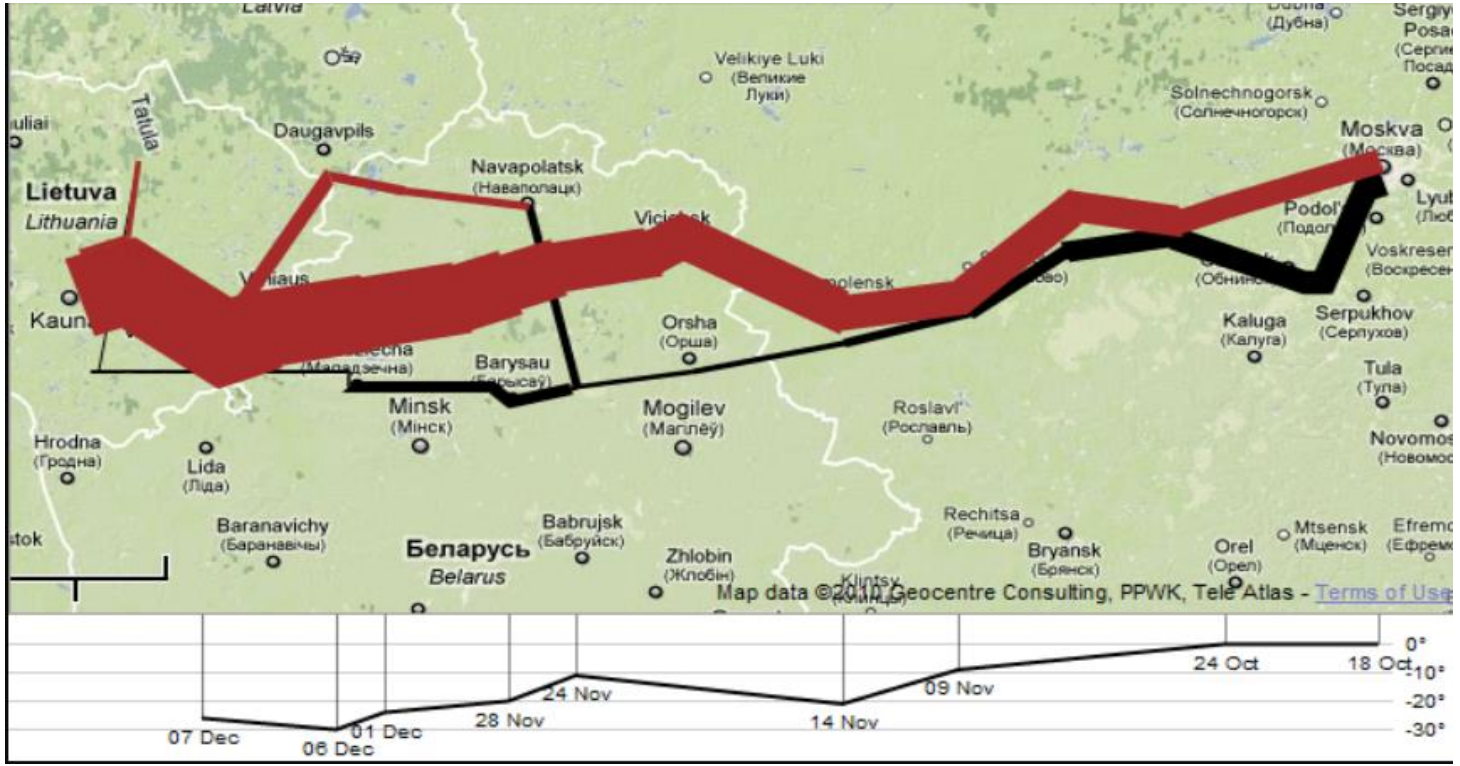


Figure: Flow map of Napoleon's March on Moscow, based on the work of **Charles Minard**
Initially - the Grand Army size was 422,000 as it progressed through Russia.

2. Choropleth Maps

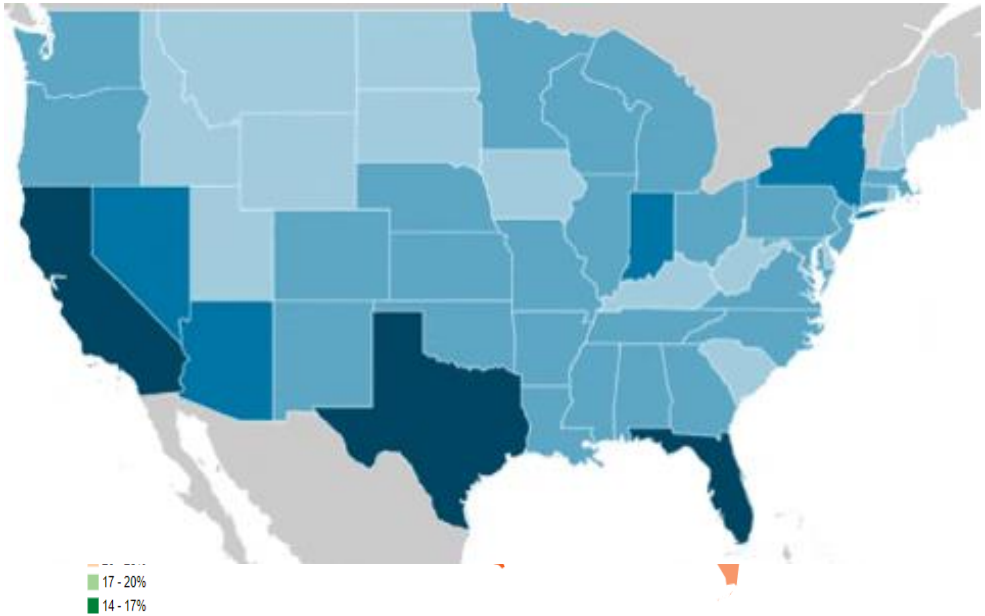


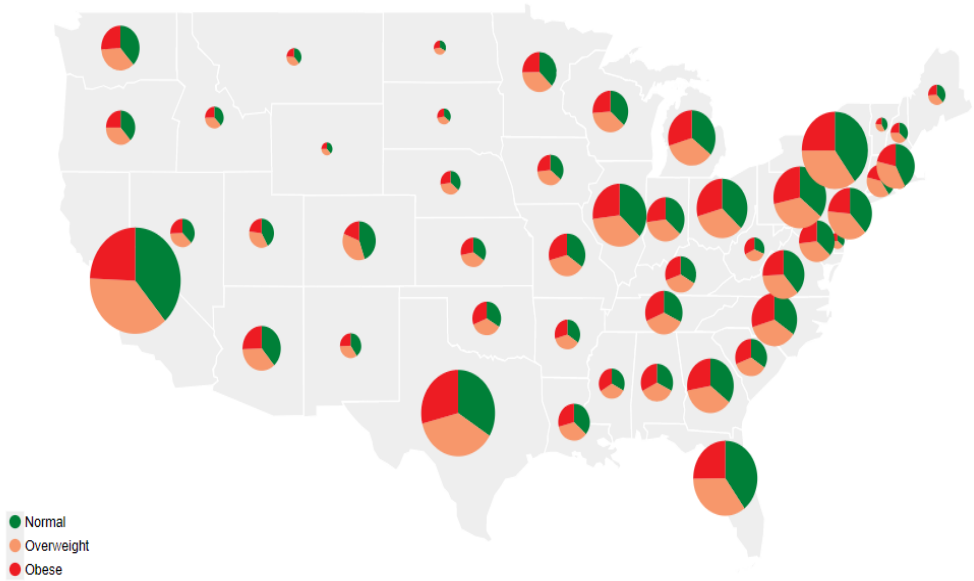
Figure: the condition of obesity in each state in the U.S.

- They represent an **aggregate summary** of a geographic characteristic within each area.
- It uses a **color encoding** approach to represent data of the geographic area.

Problems:

- Raw data is used over normalized data for density map
- Perception of shaded value can be affected by underlying area of the geographic region

3. Graduated Symbol Maps

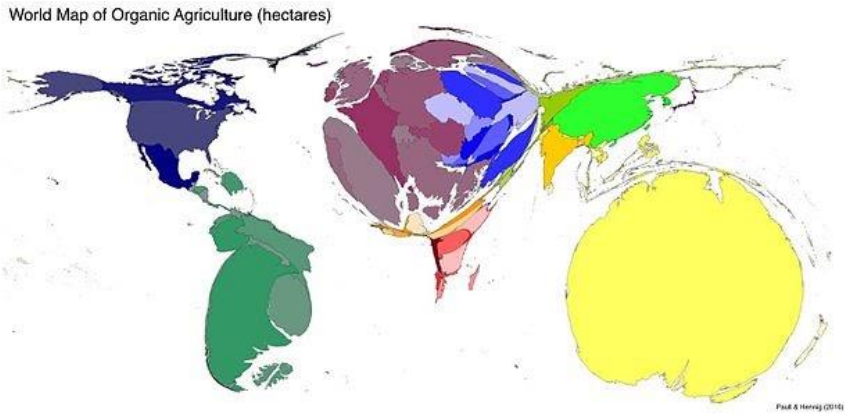


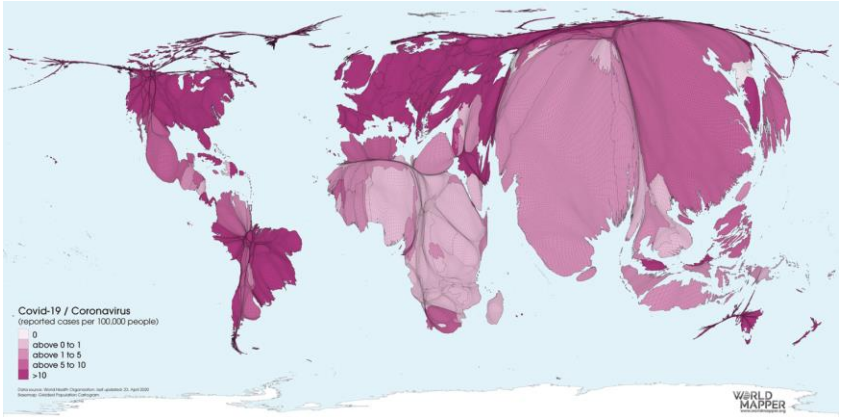
- An alternative to the choropleth map.
- Places symbols over an underlying map.
- More dimensions to be visualized i.e. symbol's size, shape and color
- Pie charts

Figure: State's population - Total circle size, Proportion of people with specific BMI rating - Each slice

4. Cartograms

- Area directly encodes a data variable
- The geometry or space of the map is **distorted** in order to convey the information



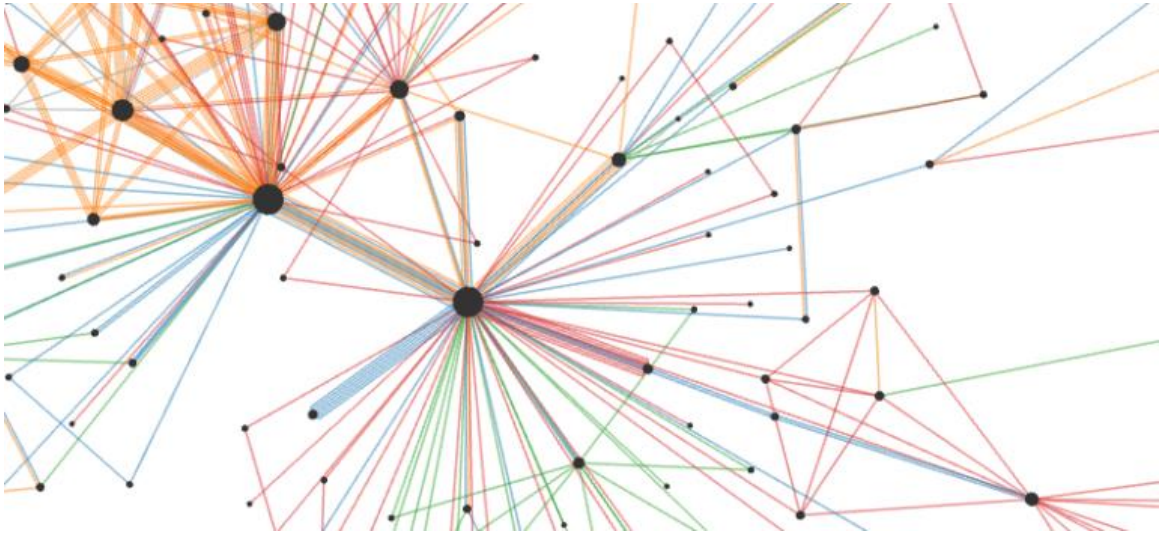


Hierarchies Visualization

- Data which is a flat collection of numbers, can be organized into natural hierarchies
- Data that can be represented:
  - Organizational Chart
  - Classifications
  - File structure
  - Decision Tree (Logical inference)

1. Node-link diagrams

- Orthogonal layout
  - Indented Layout
  - Dentrogram
  - Icicle
- Traditional Layout
  - 2D
  - 3D : Cone Trees
  - Radial layout
- Radial diagram
  - Sunburst diagram
  - Hyperbolic trees



Node-link diagrams: Traditional approach

- Allocate Space proportional to no. of Children at different Levels.
- Leaf nodes are not at same level
- **Reingold-Tilford algorithm**: Make smarter use of space maximize density space, maximize density and symmetry

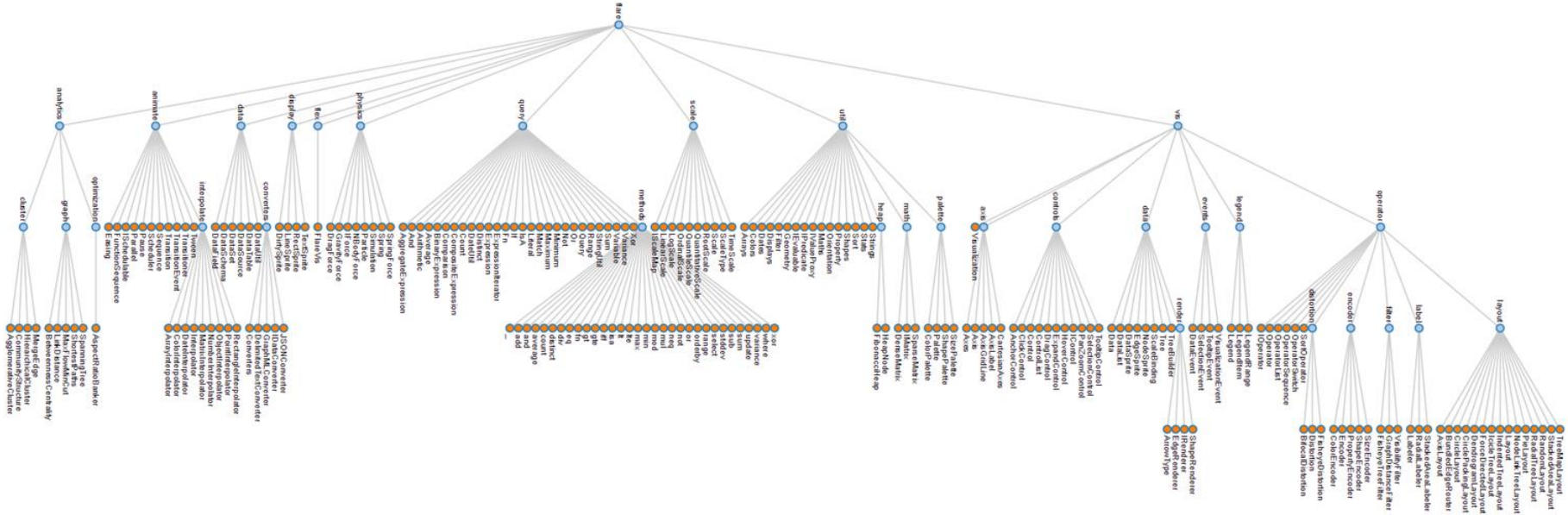
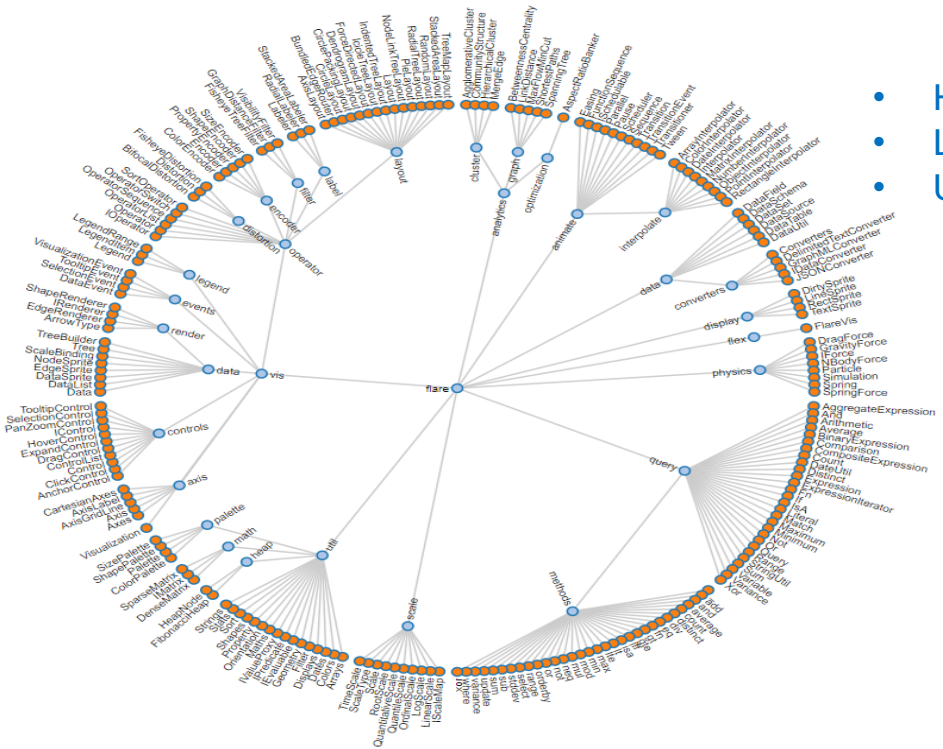


Figure: Package hierarchy of software. Reference

Node-link diagrams: Dendrogram - Radial Layout



- Hierarchy of categories based on degree of similarity
- Leaf nodes of tree at same level
- Uses space more efficiently

Figure: Classes and its packages (Reference)

Node-link diagrams: Indented tree layout

Operating system to represent the directories

Features:

- Easy to implement
- Allows multivariate data(size, last modified)
- Good for Searching, Bad for Structure

Problems:

- Requires excessive vertical space

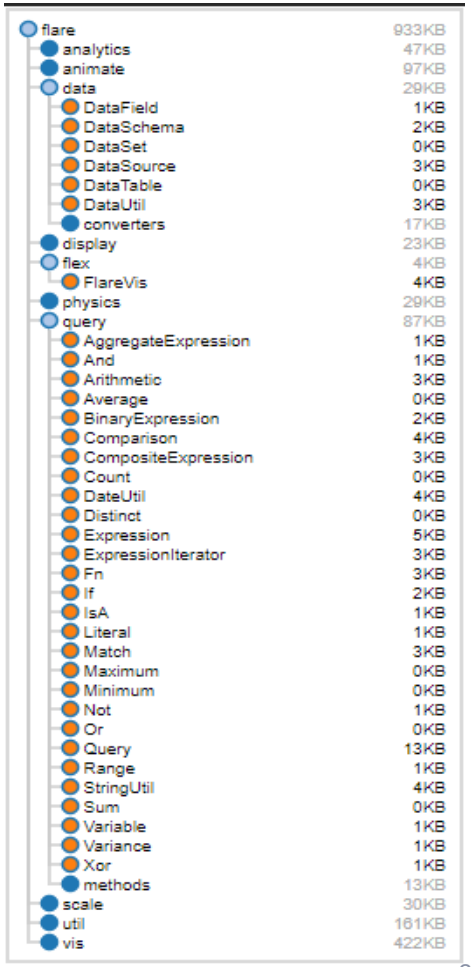


Figure: Directory of classes and packages(Reference)

Hierarchies Visualization:Adjacency Diagrams – icicle layout

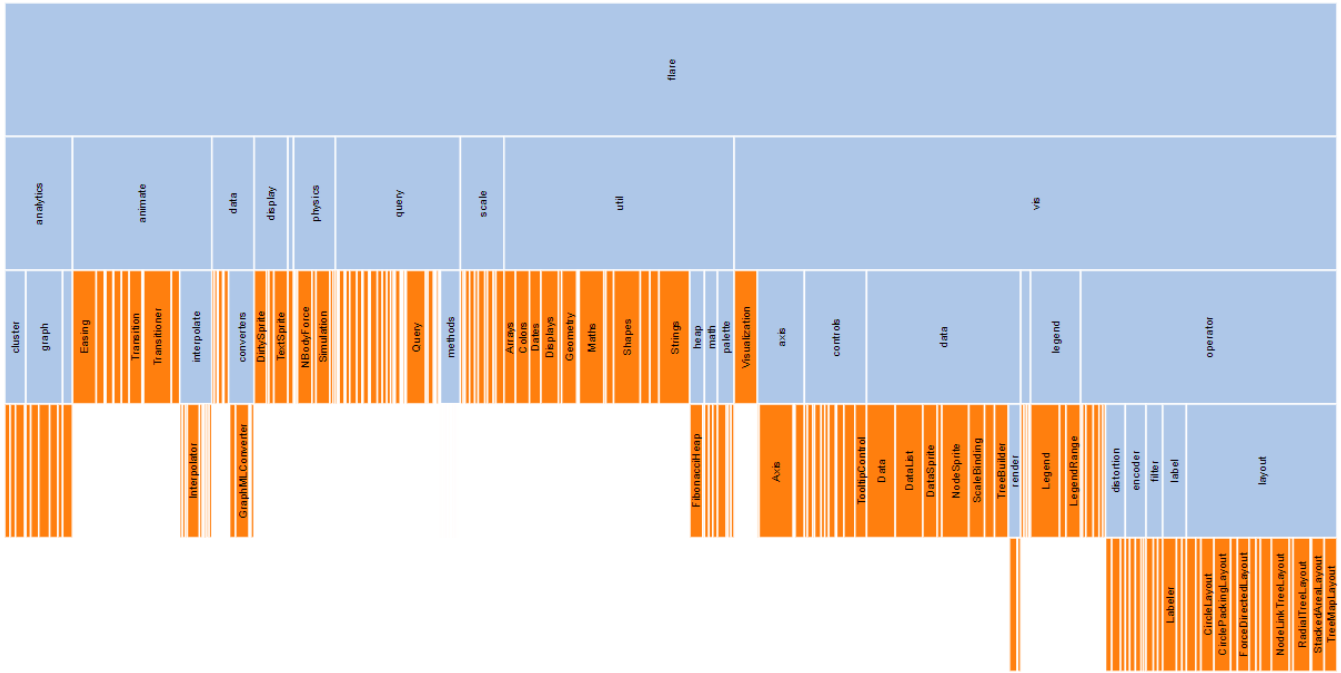


Figure: Icicle tree layout of the Flare package hierarchy[1] Reference

Hierarchies Visualization:Adjacency Diagrams – Sunburst layout

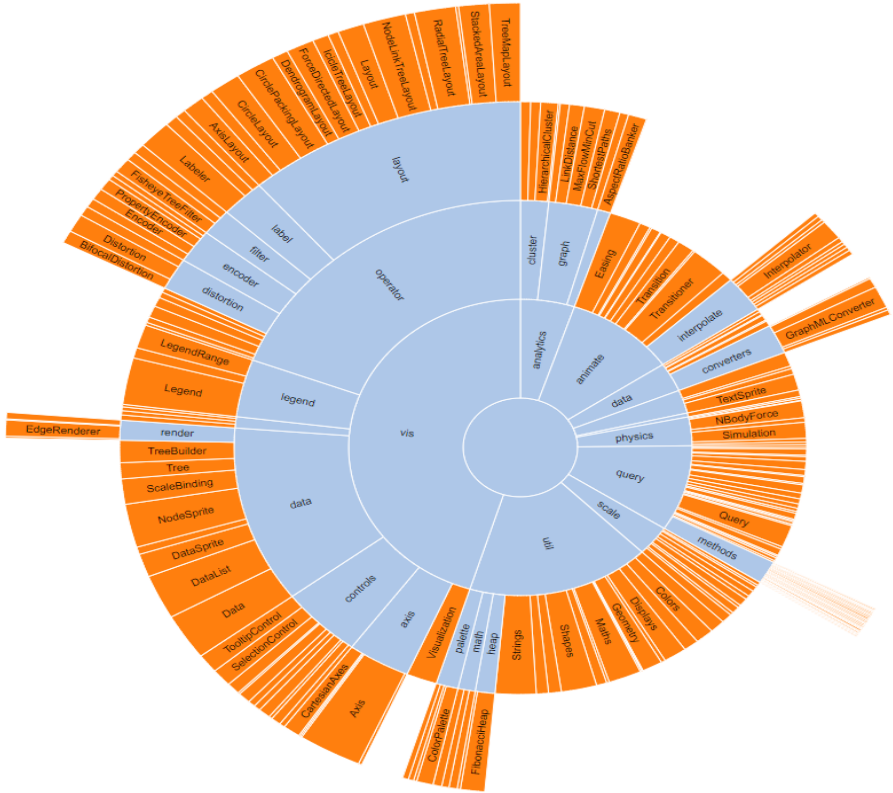


Figure: Sunburst tree layout of the Flare package hierarchy[1]. Reference

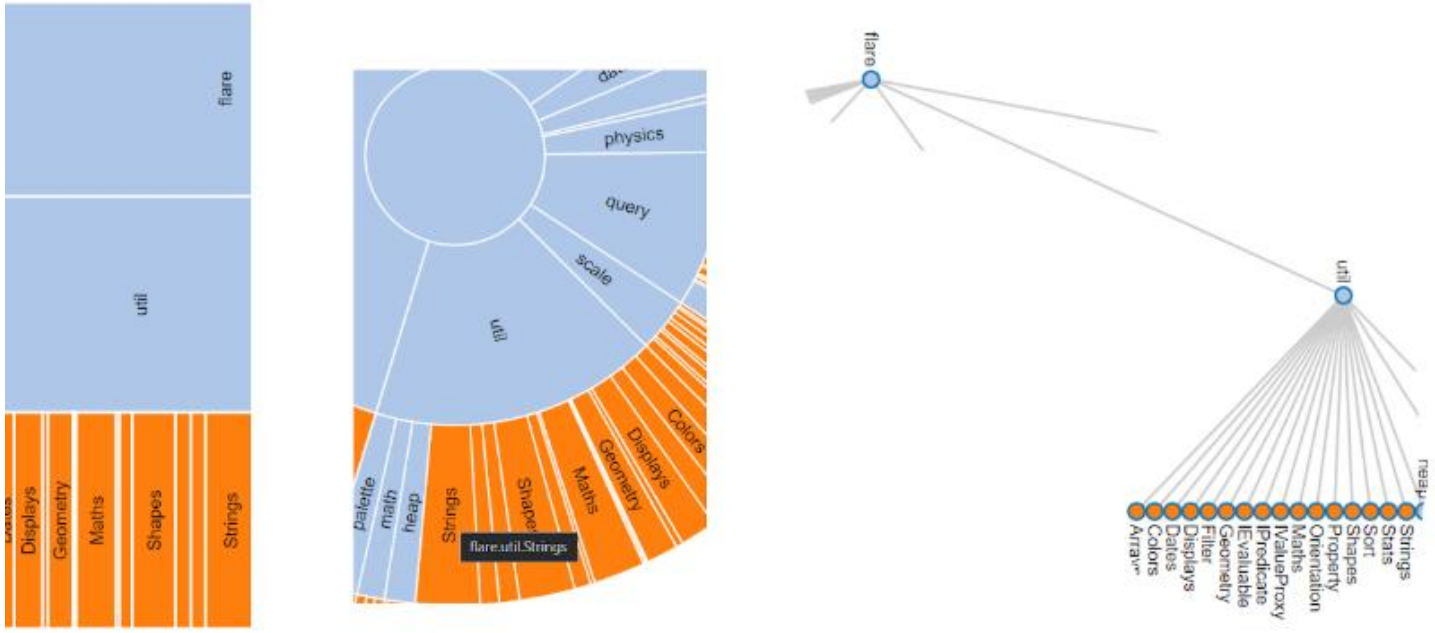
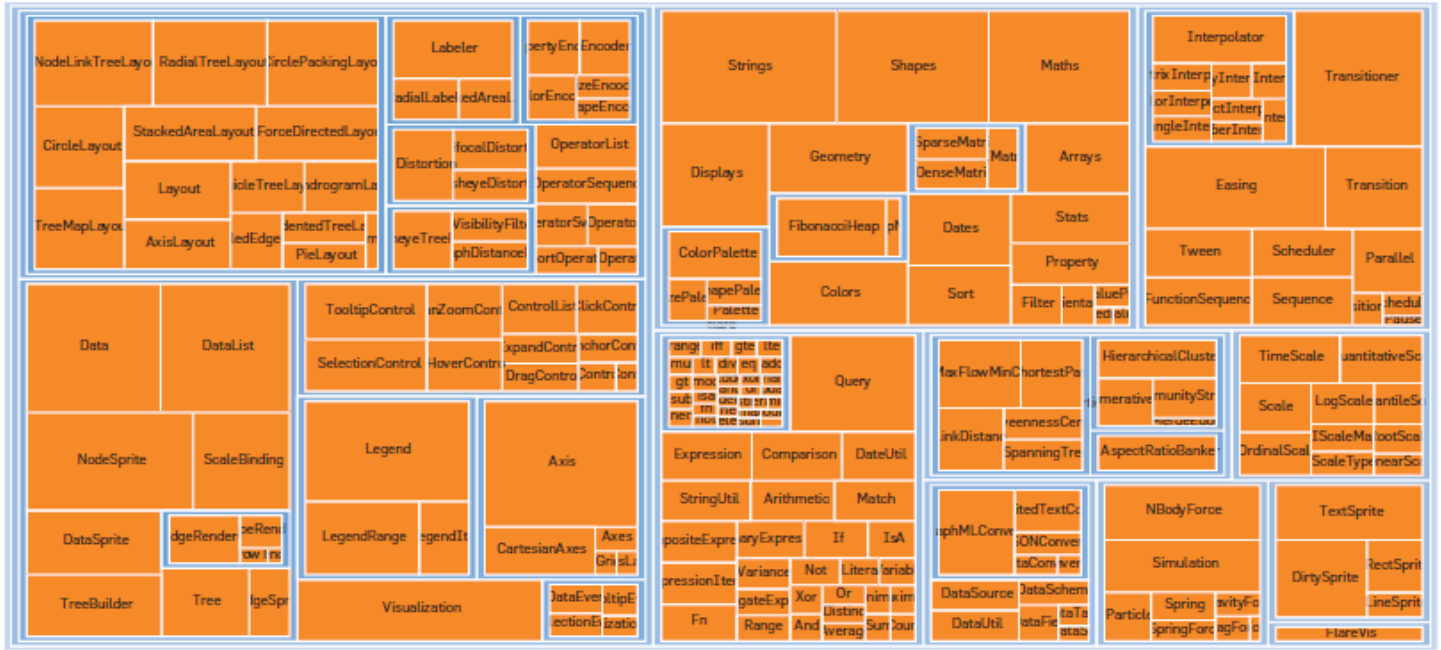


Figure a Figure b Figure c

Figure: (a)Adjacency Diagrams- Icicle layout, (b)Adjacency Diagrams- Sunburst layout (c) Node Link diagram

Hierarchies Visualization: Enclosure Diagrams-Treemap layout



http://hci.stanford.edu/jheer/files/zoo/ex/hierarchies/treemap.html

Figure: Treemap layout of the Flare package hierarchy[1].

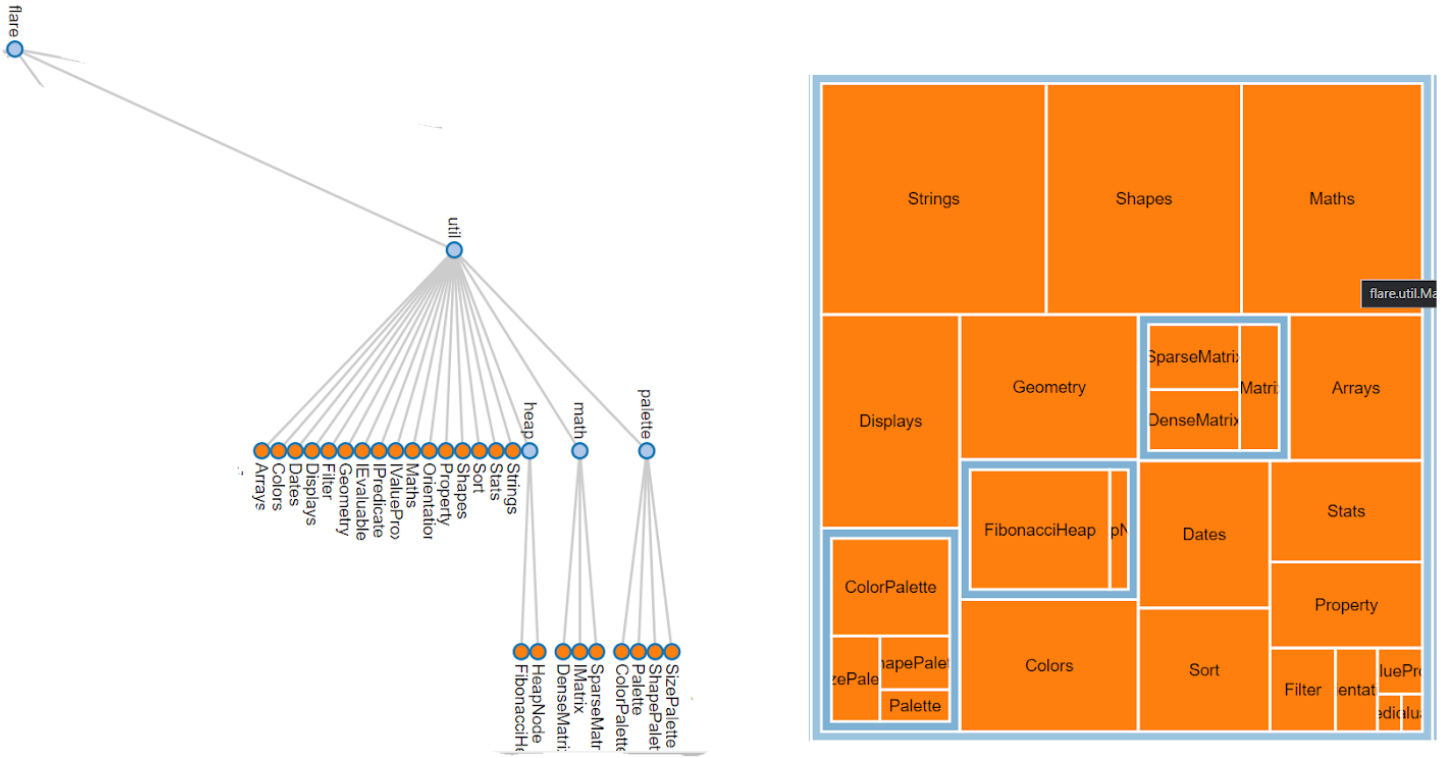


Figure: (a) Node Link diagram (b) Tree map layout

Hierarchies Visualization: Enclosure Diagrams-circle packing layout

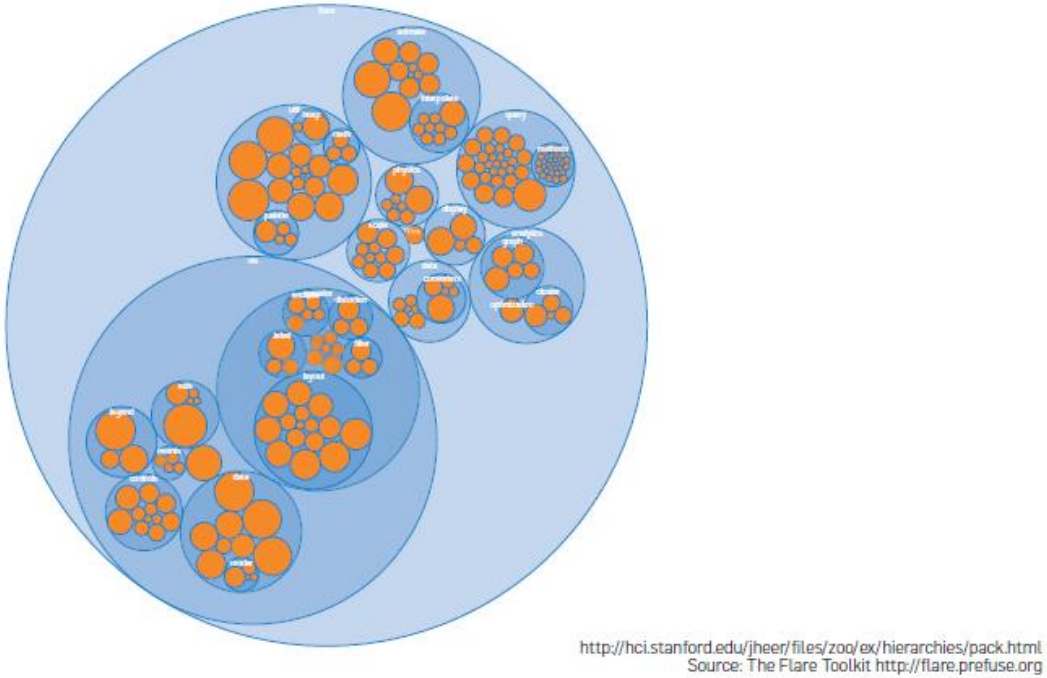
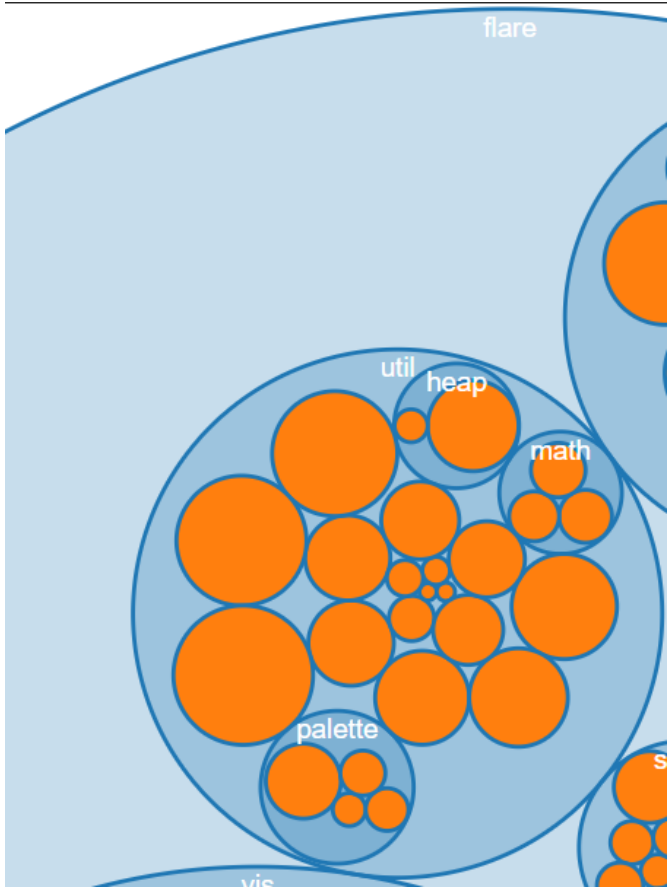


Figure: Nested circles layout of the Flare package hierarchy


flare
util
heap
math
palette

Networks Visualization

1. Force-directed Layouts
2. Arc Diagrams
3. Matrix Views

Force-directed Layouts :Physical Model

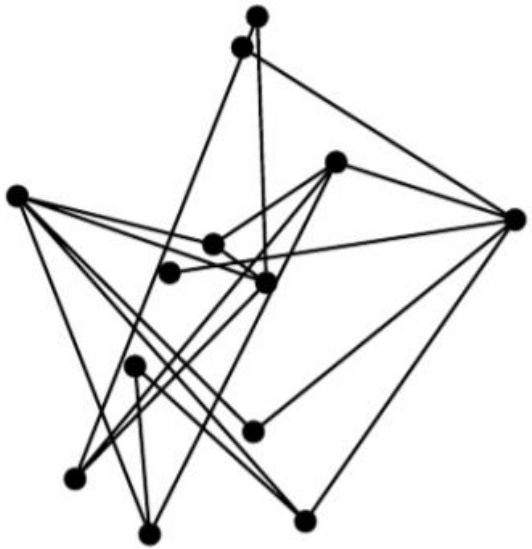


Figure: Undirected graph

Force-directed Layouts :Physical Model

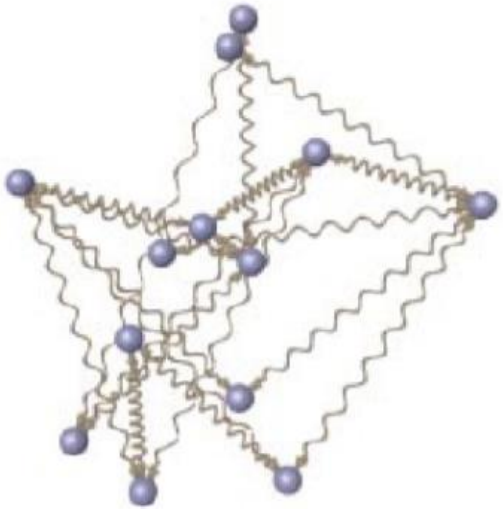


Figure: Undirected graph

“To embed a graph we replace the vertices by steel rings and replace each edge with a spring to form a mechanical system . . .

Force-directed Layouts :Physical Model

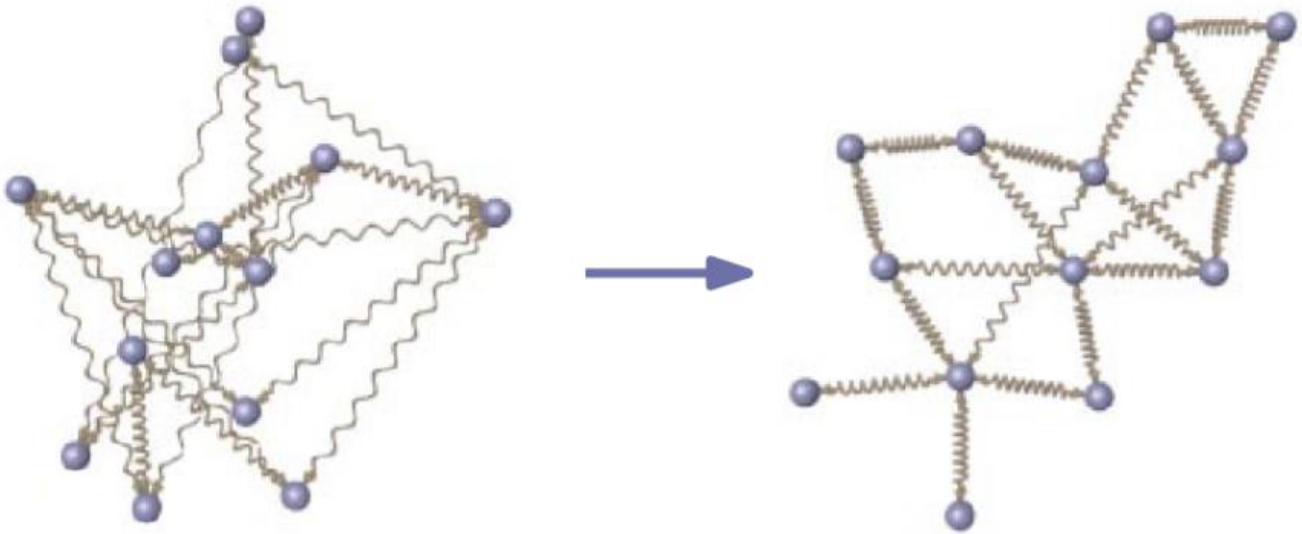


Figure: Undirected graph

“To embed a graph we replace the vertices by steel rings and replace each edge with a spring to form a mechanical system . . . The vertices are placed in some initial layout and let go so that the spring forces on the rings move the system to a minimal energy state.”

Force-directed Layouts :Physical Model

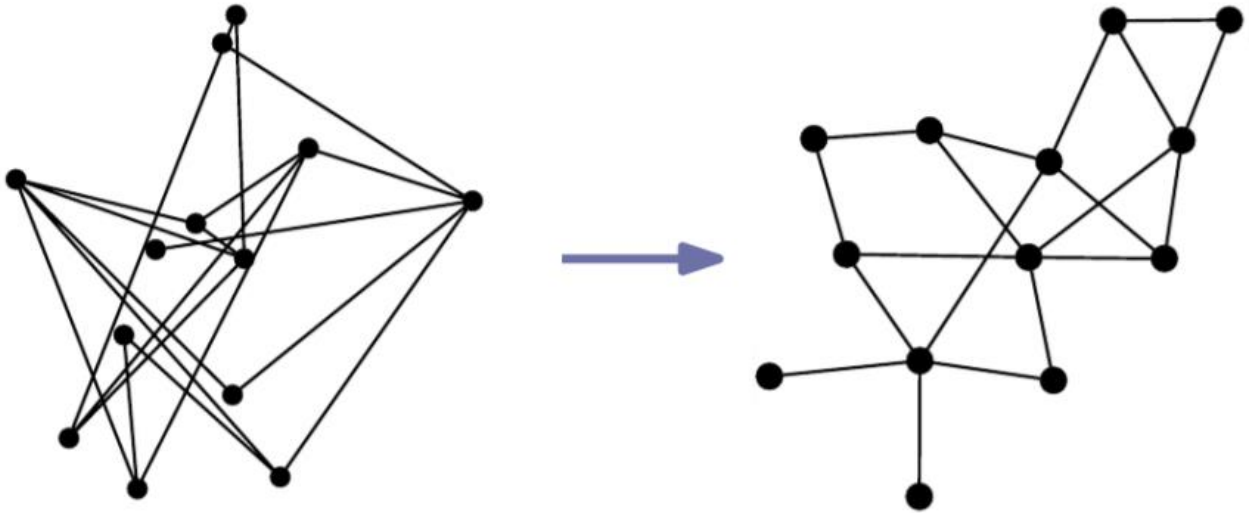
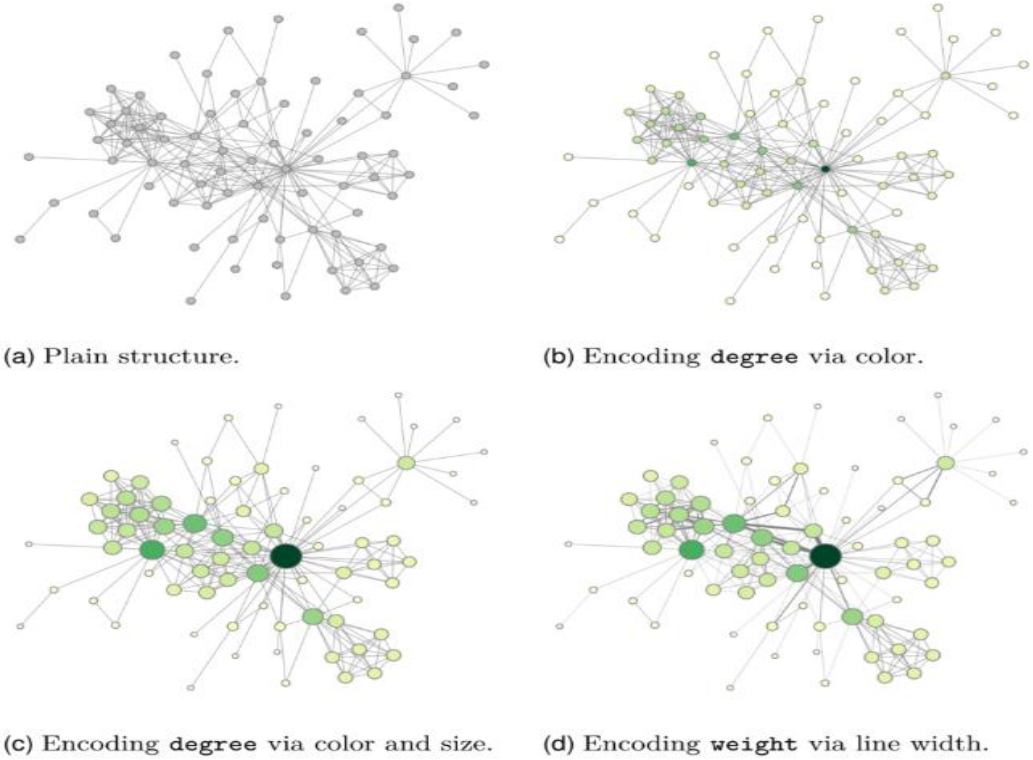


Figure: Force-directed layouts

Networks Visualization: Force-directed Layouts



(a) Plain structure.

(b) Encoding `degree` via color.

(c) Encoding `degree` via color and size.

(d) Encoding `weight` via line width.

Figure: Node-link diagram visualizing a graph structure and attributes[2].

Networks Visualization: Force-directed Layouts

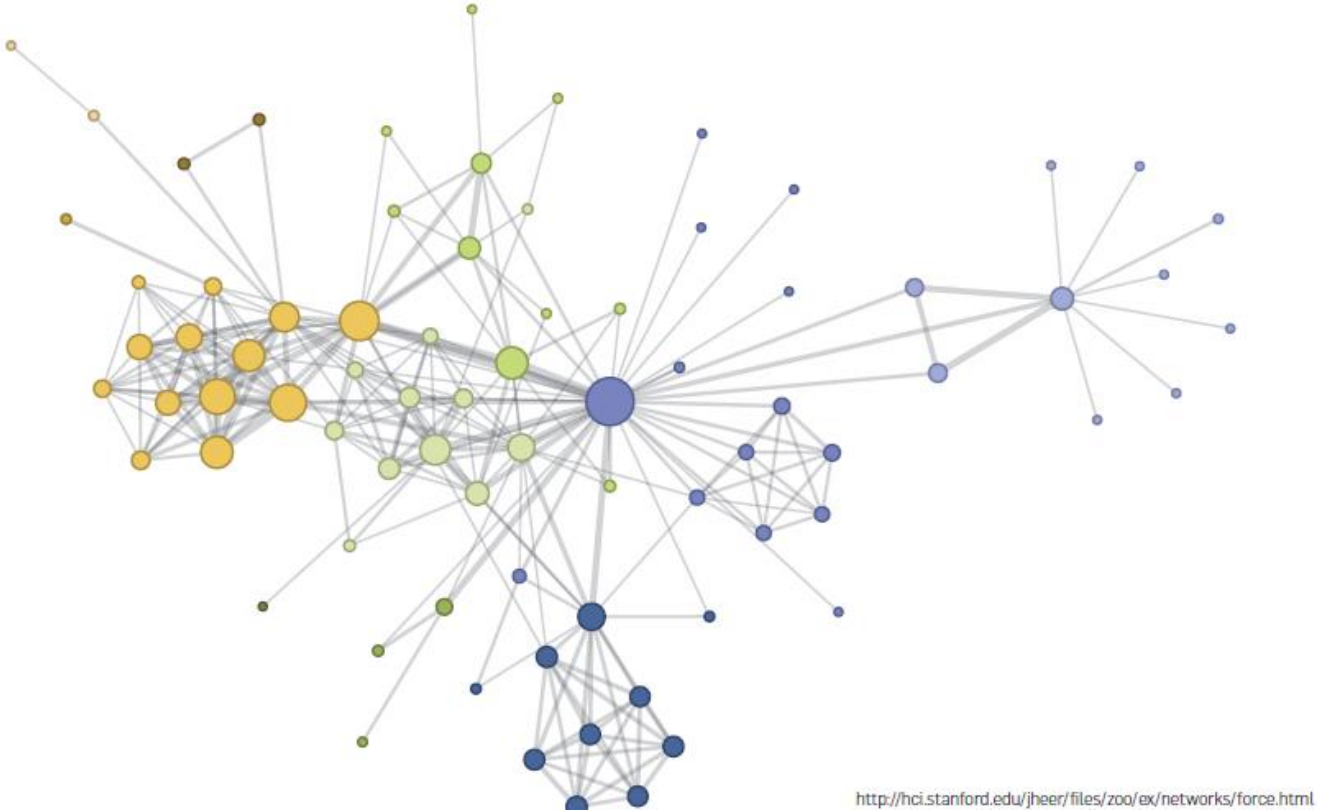


Figure: Force-directed layout of Les Misérables character co-occurrences[1].

Networks Visualization: Arc Diagrams

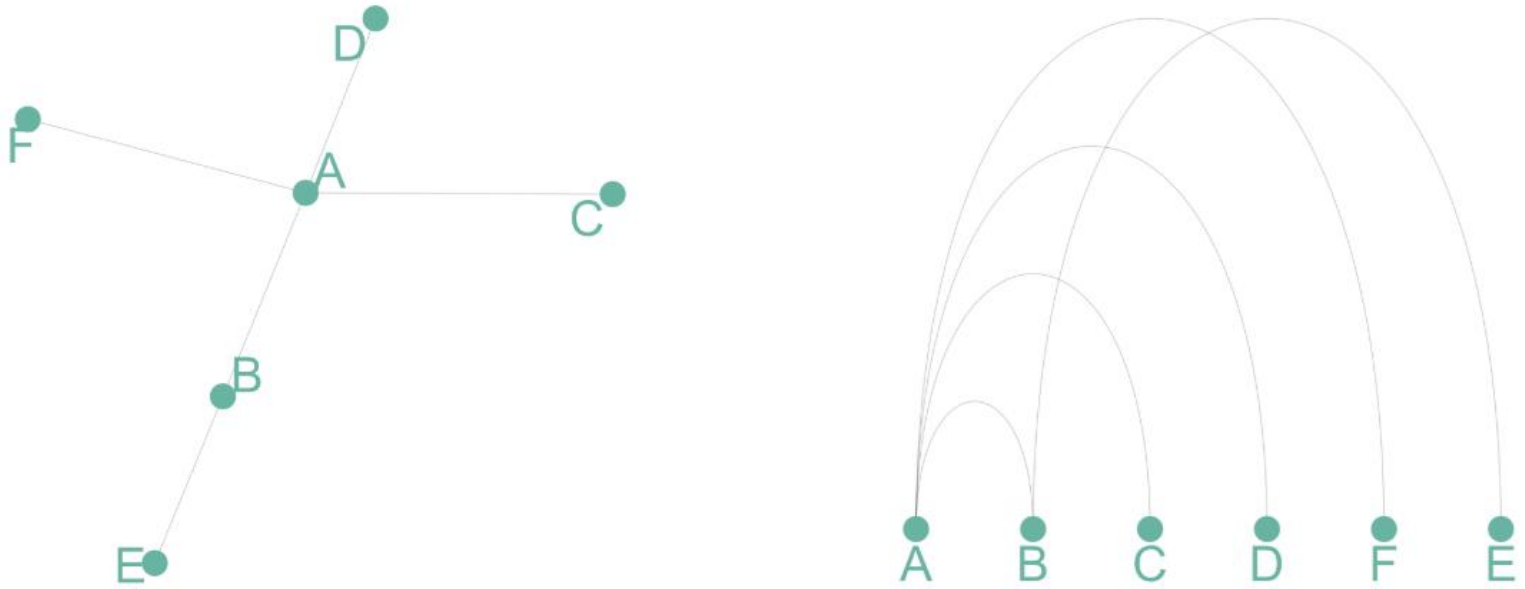


Figure: Five links between 6 nodes are represented using a **2d network diagram** (left) and an **arc diagram** (right)

Arc Diagrams

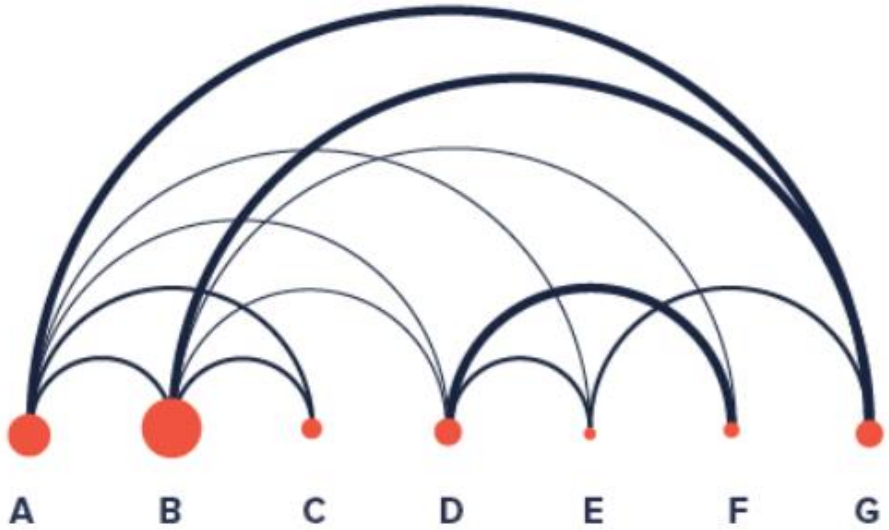


Figure: Arc Diagrams

Source credit:https://datavizproject.com/data-type/arc-diagram/

Networks Visualization: Arc Diagrams

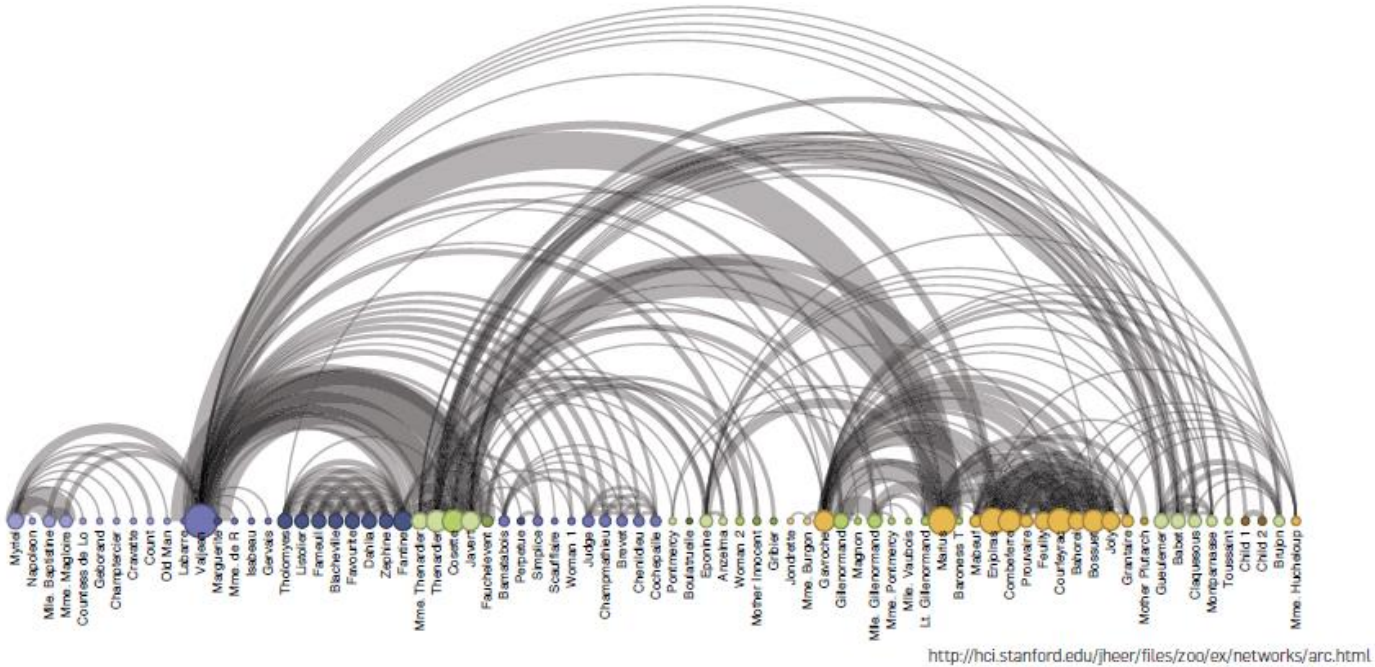


Figure: Arc diagram of Les Misérables character co-occurrences[1].

Networks Visualization: Arc Diagrams

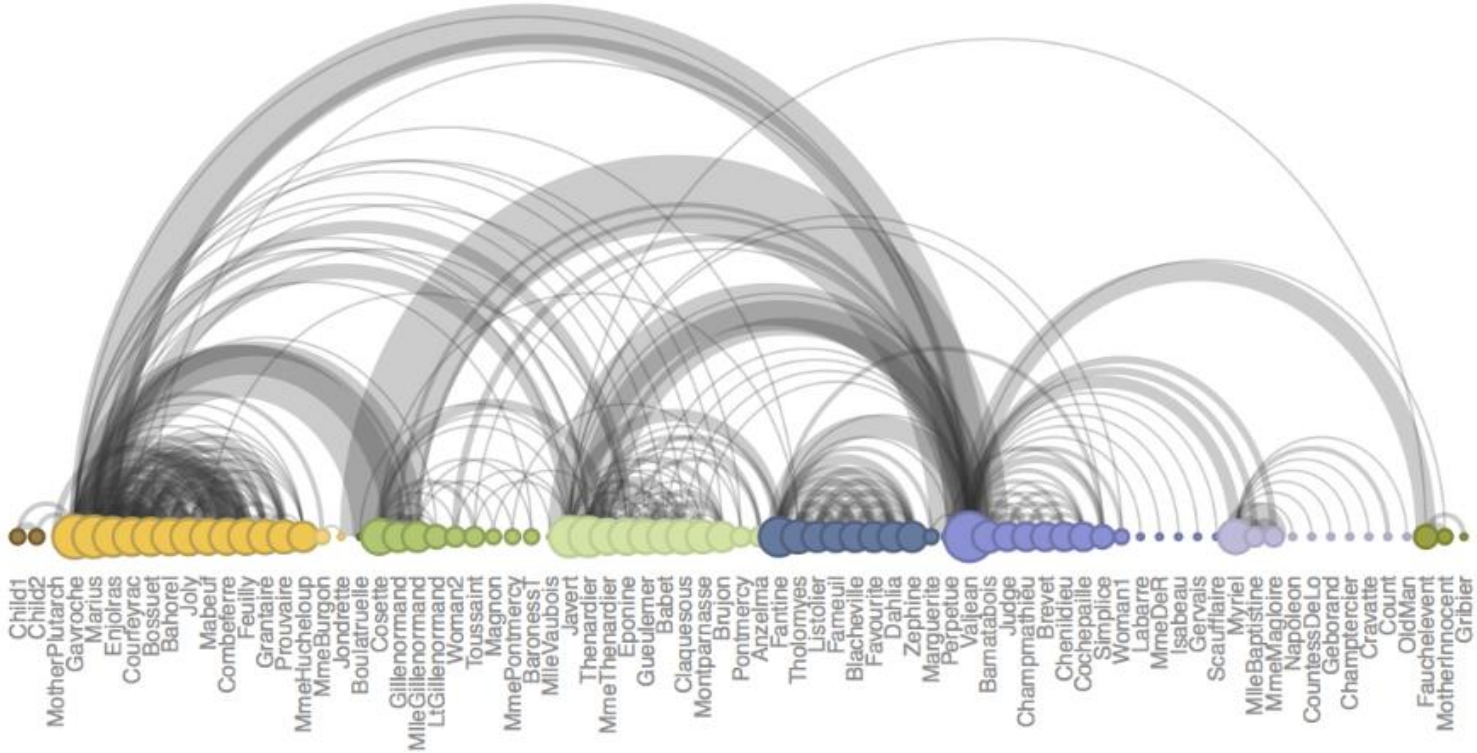


Figure: Arc diagram of Les Misérables character co-occurrences

Networks Visualization: Arc Diagrams

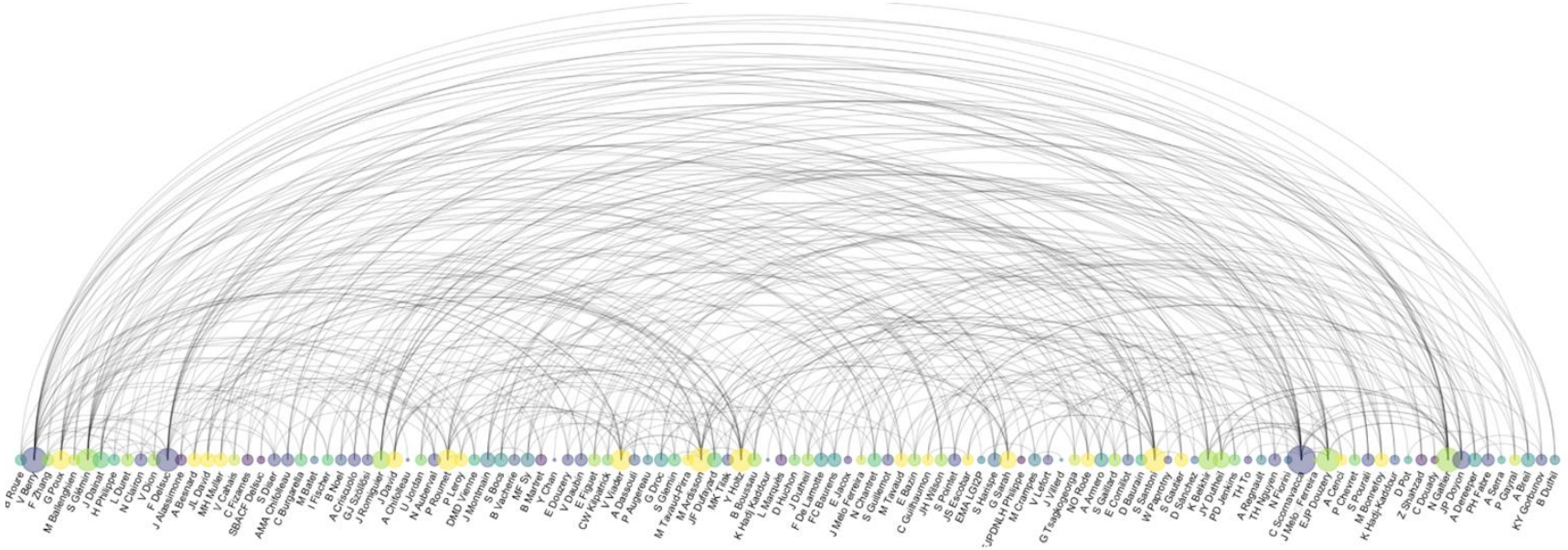
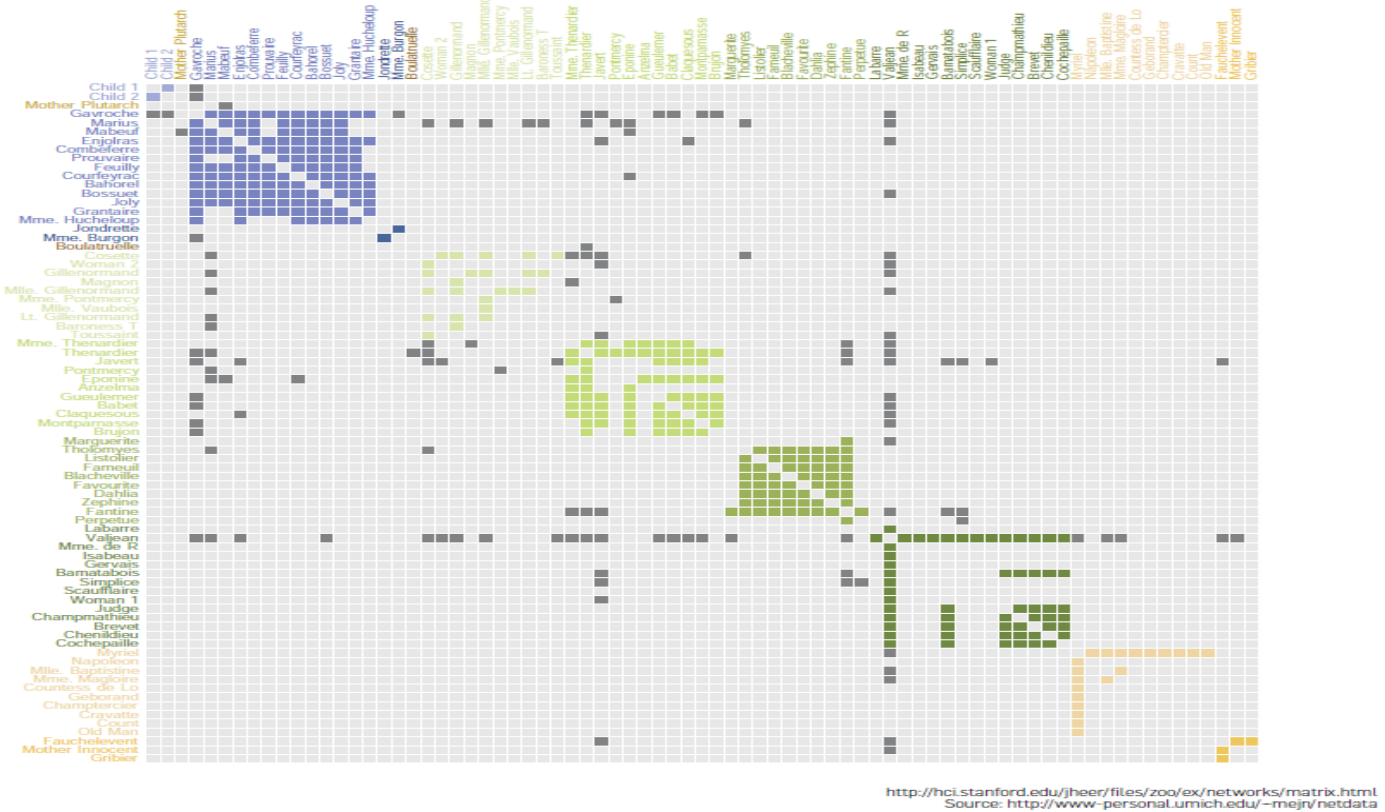


Figure: Arc diagram of Les Misérables character co-occurrences

Source credit:https://www.data-to-viz.com/graph/arc.html

Networks Visualization: Matrix Views



http://hci.stanford.edu/jheer/files/zoo/ex/networks/matrix.html
Source: http://www-personal.umich.edu/~mejn/netdata

Figure: Matrix view of Les Misérables character co-occurrences[1].

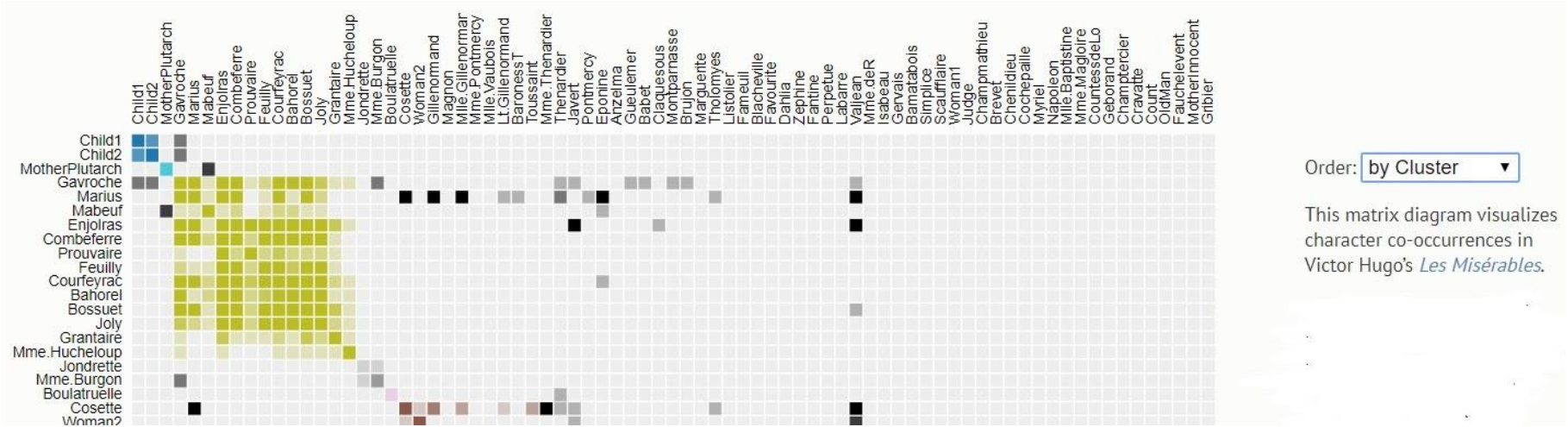
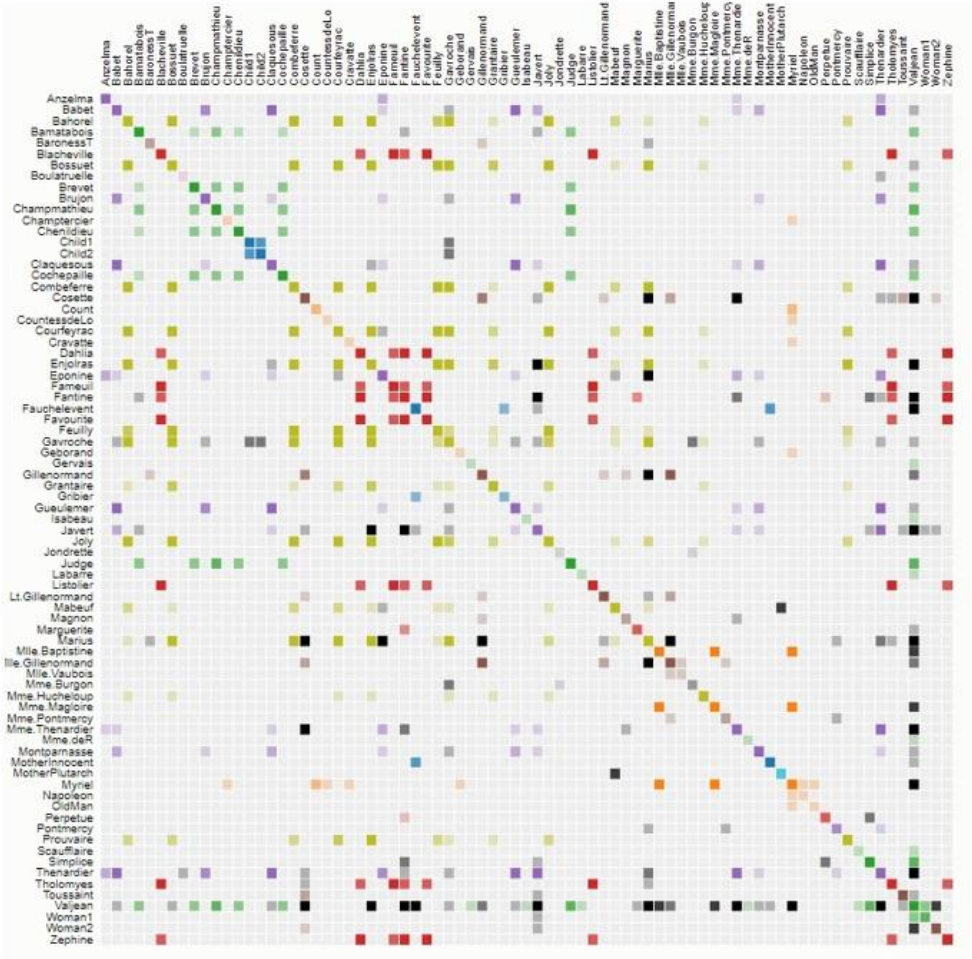


Figure: Matrix view of Les Misérables character co-occurrences[1]



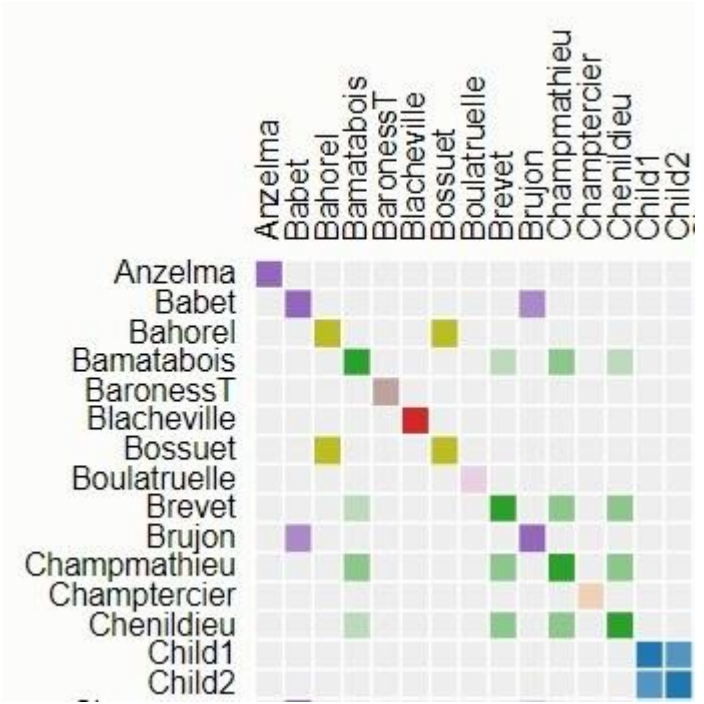


Figure: Matrix view of Les Misérables character co-occurrences[1]

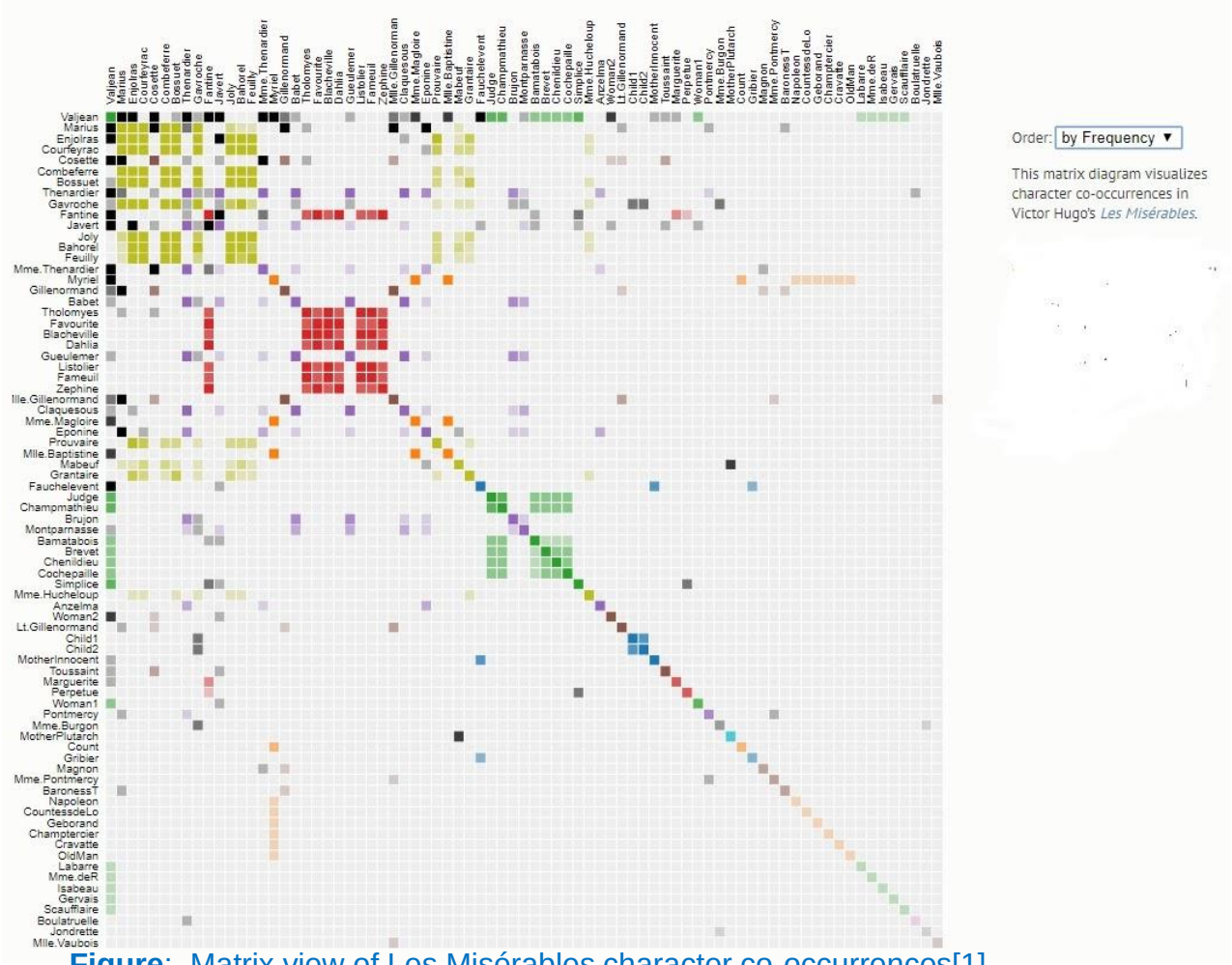


Figure: Matrix view of Les Misérables character co-occurrences[1].

Conclusion and personal critics

- All visualizations share a common “DNA ” - a set of mappings between data properties and visual attributes such as position, size, shape, and color - and customized species of visualization might always be constructed by varying these encodings.

Critics:

- Overall a well written paper with a good summarization of some of the visualization techniques.
- Paper provides a interactive webpage to all the visualization(But some of the URL is not working).
- Uses the same dataset to explain the visualization of the same category so hence comparison is possible.
- Paper sufferers from the limitations of any survey paper.

Reference

[1]. Jeffrey Heer, Michael Bostock, and Vadim Ogievetsky, “A Tour Through the Visualization Zoo” , *ACM*, 2010.

[2]. Christian Tominski, Heidrun Schumann,” Interactive visual data analysis, *CRC Press*.

[3] https://mode.com/example-gallery/force-directed-graph/

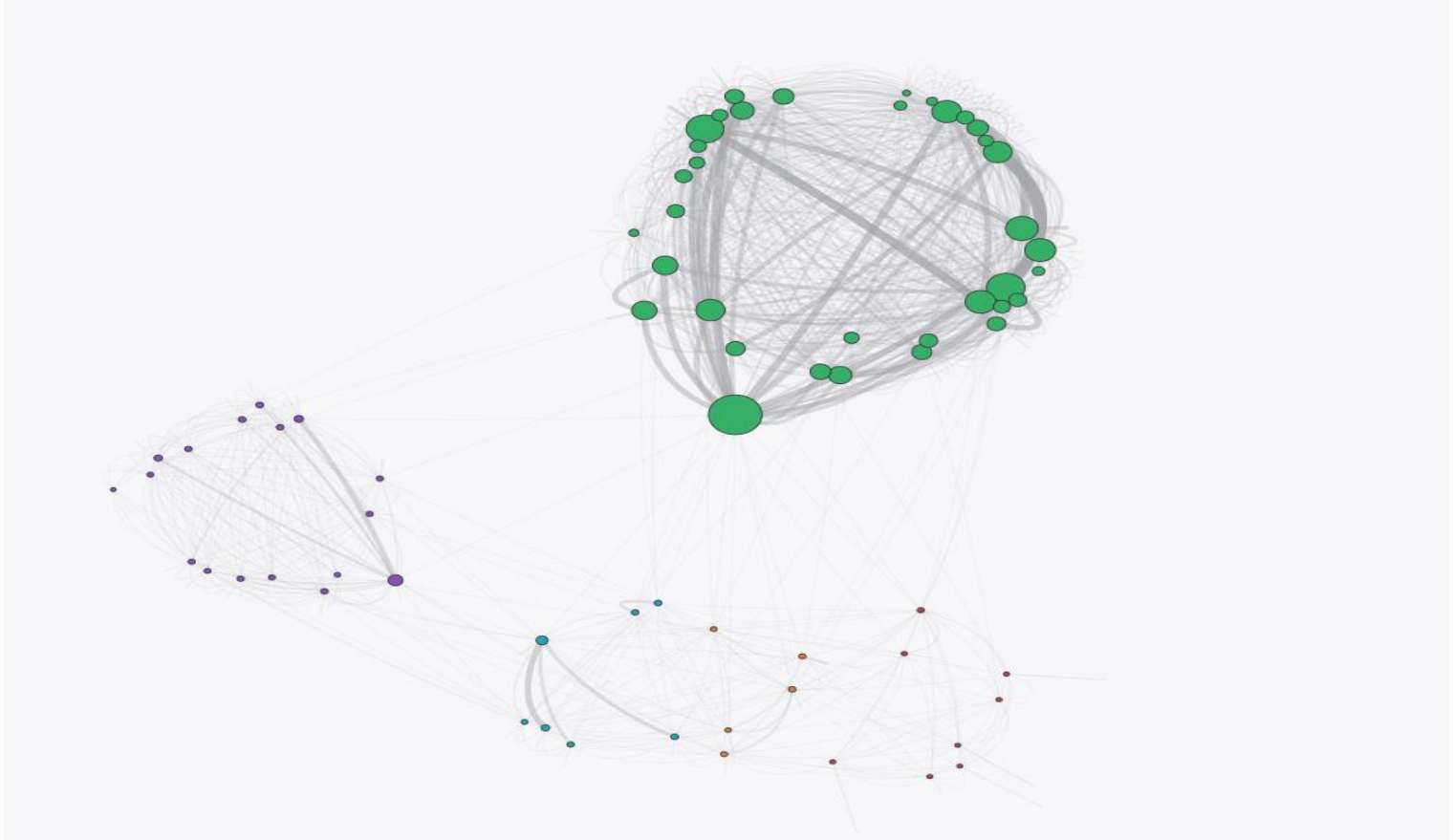
[4]. Martin Wattenberg, “Arc Diagrams: Visualizing Structure in Strings”, *IEEE*, 2002

[5] https://bost.ocks.org/mike/miserables/

Thank You!

Any Questions?

More examples:Force directed graph



Notation

$\ell = \ell(e)$	ideal spring length for edge e
$p_v = (x_v, y_v)$	position of node v
$\|\|p_u - p_v\|\|$	Euclidean distance between u and v
$\overrightarrow{p_u p_v}$	unit vector pointing from u to v

Spring-Embedder (Eades, 1984)

Model:

- repulsive force between two non-adjacent nodes u and v

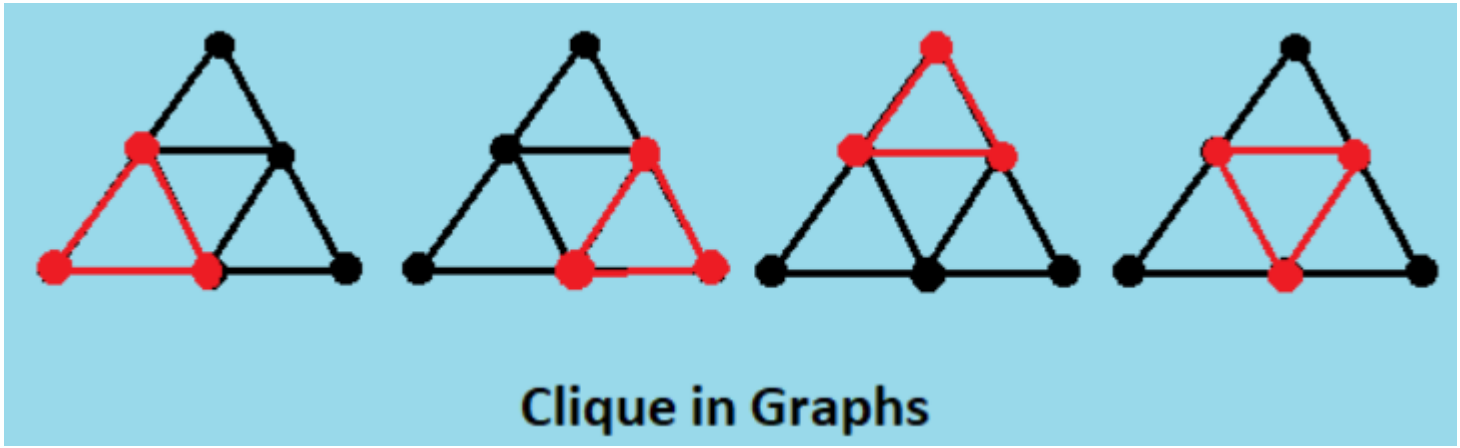
$$f_{\text{rep}}(p_u, p_v) = \frac{c_{\text{rep}}}{||p_v - p_u||^2} \cdot \overrightarrow{p_u p_v}$$

- attractive force between adjacent vertices u and v

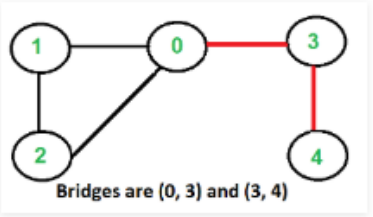
$$f_{\text{spring}}(p_u, p_v) = c_{\text{spring}} \cdot \log \frac{||p_u - p_v||}{\ell} \cdot \overrightarrow{p_v p_u}$$

- resulting displacement vector for node v

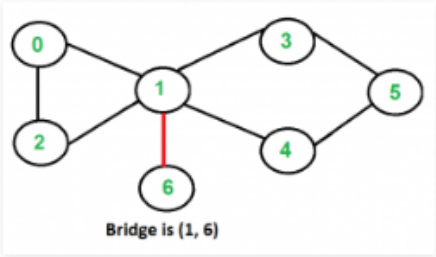
$$F_v = \sum_{u:\{u,v\} \notin E} f_{\text{rep}}(p_u, p_v) + \sum_{u:\{u,v\} \in E} f_{\text{spring}}(p_u, p_v)$$


Clique in Graphs

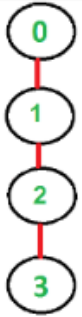
Bridges



Bridges are (0, 3) and (3, 4)



Bridge is (1, 6)



Bridges are (0,1), (1,2) and (2,3)