CS732/DS732: Data Visualization Assignment 3
Information visualization Report

Daggubati Siri Chandana - MS2019005

November 2019

1 Problem Statement

Demonstrate treemap, sunburst, node-link diagram, matrix visualization, and parallel coordinates plot visualization along with appropriate user interactions. Use D3.js/Processing to implement hierarchical and multivariate data visualizations and Gephi as a network visualization tool to analyze network data.

1.1 Data Description

The data sets used for this assignment are:

- **Multivariate(AutoMPG):** http://mlr.cs.umass.edu/ml/datasets/Auto+MPG

  The data file is a slightly modified version of the original dataset. In the modified data set, eight instances were removed because they had unknown values for the "mpg" attribute. In data, there are three multivalued discrete(cylinders, model year, origin) and five continuous(mpg, displacement, horsepower, weight, acceleration) attributes.

- **Network(Jazz Musicians):** http://konect.uni-koblenz.de/networks/arenas-jazz

  This data is the collaboration network between Jazz musicians, which is collected in 2003. Each node is a Jazz musician, and an edge denotes that two musicians have played together in a band. The data is undirected, unweighted, and only has edge information.

- **Hierarchical:** https://www.kaggle.com/fernandol/countries-of-the-world/download

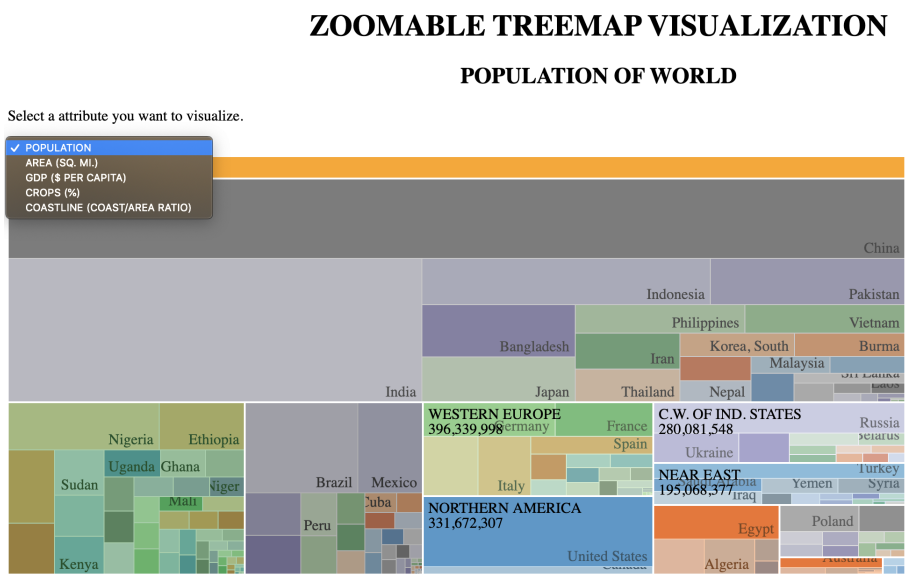
  The hierarchical data set are countries of the world data set which follows hierarchy based on region, and we have some information on population, region, area size, infant mortality, and more. The source of this data set is from the Data from the World Factbook is the public domain link.

2 Hierarchical Data Visualization

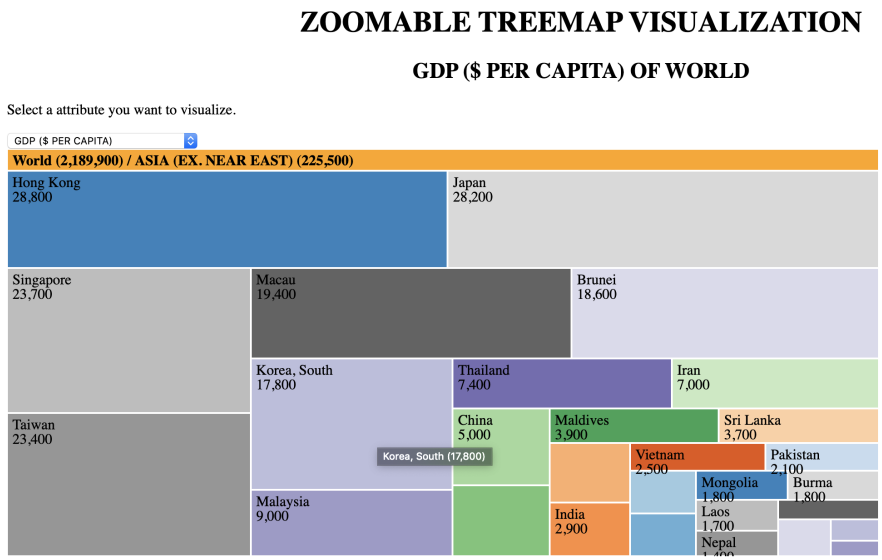
Hierarchical Visualization methods that show how data or objects are ranked and ordered together in an organization or system. Here we perform Treemap and Sunburst Diagrams on the countries of world data following the hierarchy of countries based on region. We visualize a few attributes like Population, Area (sq. mi.), GDP ($ per capita), Crops (%), and Coastline (coast/area ratio) using the drop-down menu among countries.

2.1 TreeMap

Treemaps are a more compact way of visualizing the hierarchical structure of a tree diagram. It helps us to compare the proportions of data between categories via their area size. It displays quantities under each category via rectangular area size, which is the proportion of quantity with other quantities in the same parent category and subcategory rectangles nested in it. The tiling algorithm is used to order and divide the rectangles. But to the darker side, treemap doesn't show the hierarchal levels as clearly as the basic Tree Diagram or Sunburst Diagram.



(a) Population of World

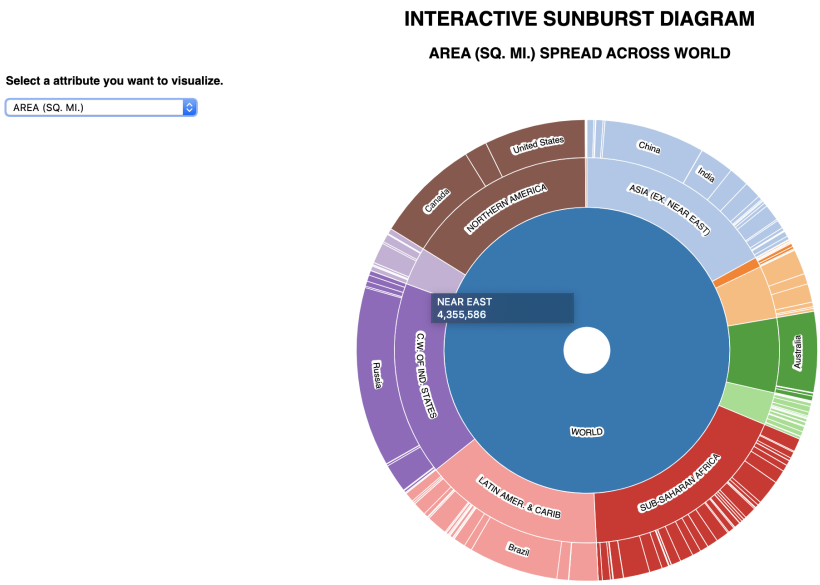


(b) GDP of NearEast Asia Region
Figure 1: Zoomable TreeMap Diagram on Countries of World

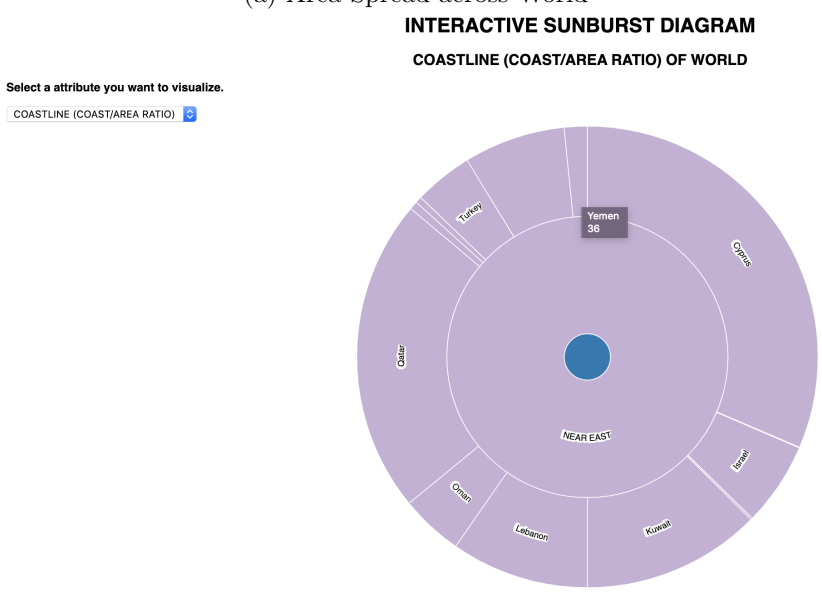
User Interactions on Treemap: Here we implemented a more interactive treemap, which shows the item and its data value when the mouse hovers on the rectangular area, and we can zoom a region to fit the entire map to study the proportions of subregions within the region.

2.2 Sunburst Diagram

The Sunburst Diagram or multi-level pie chart is an alternative, area filling visualization that uses a radial layout. The hierarchy regions depicted by concentric circles. The center of the circle is a root node, and a segment of the inner circle bears a hierarchical relationship to those segments of the outer circle which lie within the angular sweep of the parent segment. The color of arc corresponds to some attribute of the data.



(a) Area Spread across World



(b) Coast to Area ratio of NearEast Region

Figure 2: Interactive Sunburst Diagram on Countries of World

User Interactions on Sunburst Diagram: Here the interaction mechanisms of the sunburst are the same as a treemap, but it is on circles rather than rectangles.

3 Multivariate Data Visualization

Multivariate datasets with typically more than two attributes are hard to visualize since two variables restrict our data analysis. In Multivariate visualization, Parallel Coordinates and Scatterplots remain one of the widely used visual representations of them all.

3.1 Parallel Coordinates

Parallel Coordinates Plots is closely related to time series visualization, except that it is applied to data where axes points don't correspond to time. These are ideal for comparing many variables together and seeing the relationships between them. In plot, a set of parallel axes is drawn, each representing a variable. Each axis can have a different/uniform scale. Values are plotted as a series of lines that connected across all the axes. In other words, parallel coordinates plot is a collection of polylines where intersection points of axis lines and polylines are data values of variables for each data item in the dataset.

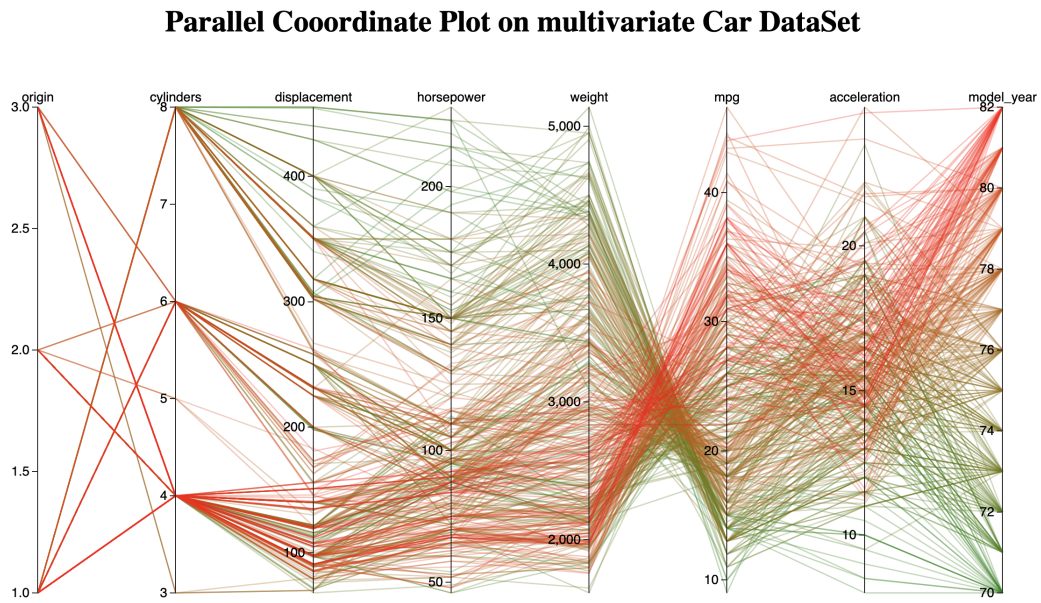


Figure 3: Parallel Coordinates on Cars Dataset

User Interactions on Parallel Coordinates Plot: This plot is made more flexible by allowing the user to change the order of axes lines according to the relationships among the variables the user wants to visualize, and the user is allowed to brush a set of ploy lines to observe their properties separately without visual clutter.

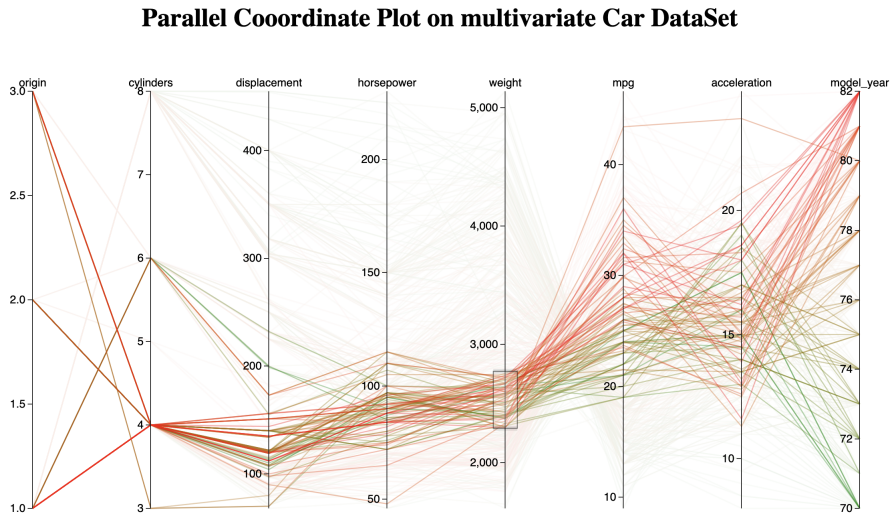


Figure 4: Brushing a set of lines on Plot

Q1. Compare the visualizations of the raw and remodeled data. What semantics of the data did you exploit for the visualization?

The Countries of World Dataset used for hierarchical visualization has multiple attributes without any hierarchy other than country and region attributes. So we are considering the point that large no attributes having no hierarchy and the multivariate nature of attributes based on the key-value country. We visualize Countries of World Dataset using parallel coordinates plot on remodeled data formed after excluding the region attribute. We also preprocess the data before performing these two visualizations, like replacing missing values with zero and commas with decimal points to get the correct result. We can get more information form parallel coordinates plot, such as relationships between attributes that are not seen in hierarchical visualization even if it is made interactive.

We can also represent the Countries of World data set using a choropleth map, which makes the visualization more user understandable as the information regarding the data set is color-coded on the world map. It indicates the intensity of data value with the corresponding color on the polygon area representing the country. Although even this doesn't give relationships between attributes like parallel coordinates, it is more meaningful than hierarchical visualization as is on Countries of World.

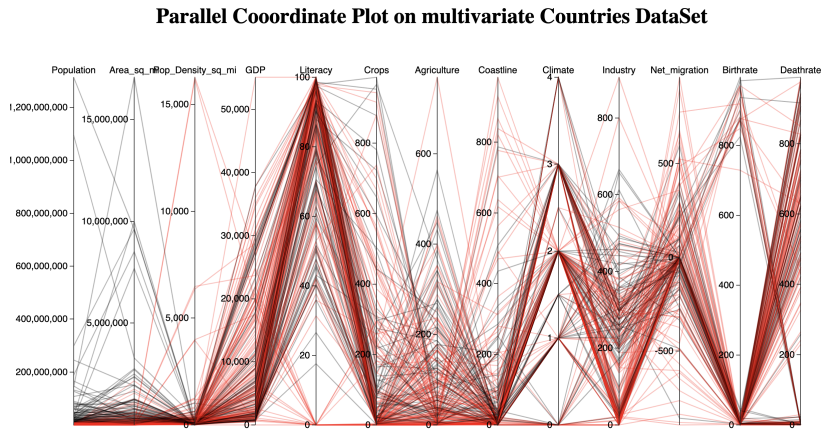


Figure 5: Parallel coordinate plot on the remodeled countries of world data set.

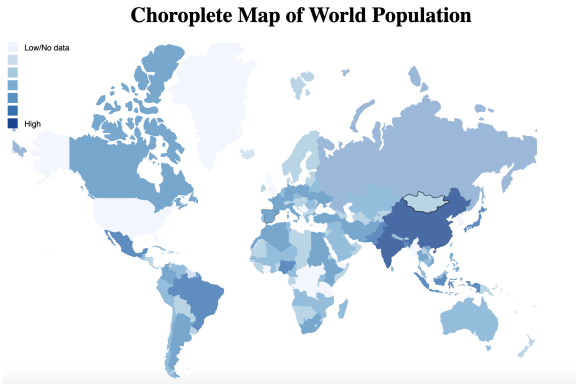


Figure 6: Choropleth Map on the remodeled Countries of World data set.

3.2 ScatterPlot Matrix

Scatterplot Matrix visualize multidimensional datasets by assigning data dimensions to graphical axes and data cases as points in the cartesian space defined by the axes. In other words, the data point in the matrix is a scatter plot. Unlike parallel coordinates, we get the relationship of each attribute with every other attribute without any user interaction. **User Interactions on Scatterplot Matrix:** To make this interactive brushing technique allowed on scatter plot points. Here we color scaled the data based on car name attribute.

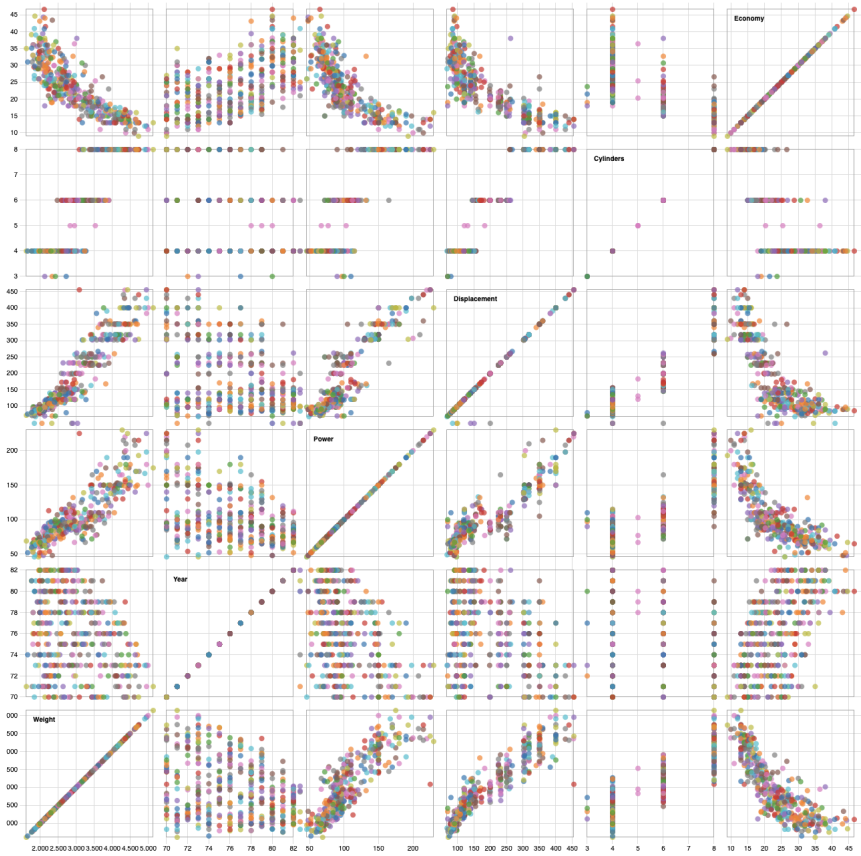


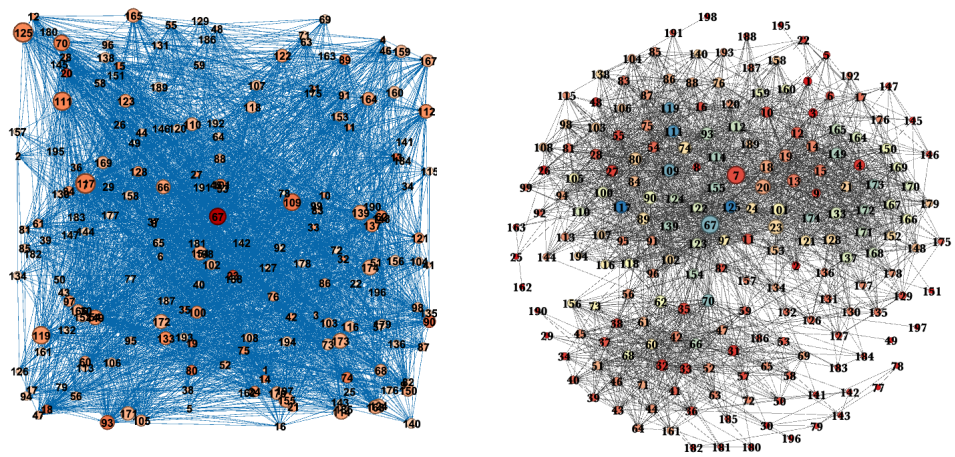
Figure 7: Scatter Plot Matrix Visualization on Car Dataset

4 Network Data Visualization

Network Visualisation or Network Graph is often used to visualize complex relationships between massive data. A network visualization displays undirected and directed graph structures. A network is a collection of data where the entities within that data are related through the principles of either connection or containment.

4.1 Node-Link Diagram

Graphs are frequently drawn as node-link diagrams that illuminate relationships between entities, which are nodes and links, respectively. Nodes nothing but vertices are represented as disks, boxes, or textual labels. Links nothing but edges are represented as line segments, polylines, or curves in the plane connecting vertices.



(a)No layout applied (b)force-directed layout algorithm

Figure 8: Node-link Diagram on Jazz Musicians Network

Q2. How did you use Gephi for network visualization and analysis?

Gephi helps us visualizing and analyzing large network graphs using a 3D render engine. Here we load the node list of the graph, then the following things are done in gephi, such as creating a data table, and we can manipulate the data and perform all data manipulation and data definition operations on the data table in the data section. Secondly, it creates a default graph for that data without the implementation of any algorithms. Now after the graph is created, we can apply layout algorithms on the graph to make visualization better. In Jazz musicians, network data force-directed layout is giving better results. Then we can also add graphical properties to elements in the graph based on some ranking criteria making visualization look better. As it is an unweighted graph, links are given uniform color, nodes are given with three color scale based on the degree of a node, and size property is applied based on in-degree of a node. The default and force-directed layout visualizations are shown in Fig. 8.

4.2 Adjacency Matrix Visualization

An adjacency matrix is a square matrix used to represent a finite graph. The elements of the matrix indicate whether pairs of vertices are adjacent or not in the graph. Here we have an unweighted and undirected graph where the adjacency matrix is a zero diagonal symmetric matrix with zeros representing no edge and ones representing the edge.

User Interactions on Adjacency Matrix: Here when the mouse hovers over the cell in the adjacency matrix, it highlights the row and column of the corresponding cell in the matrix enabling the user to get the x-axis and y-axis labels quickly.

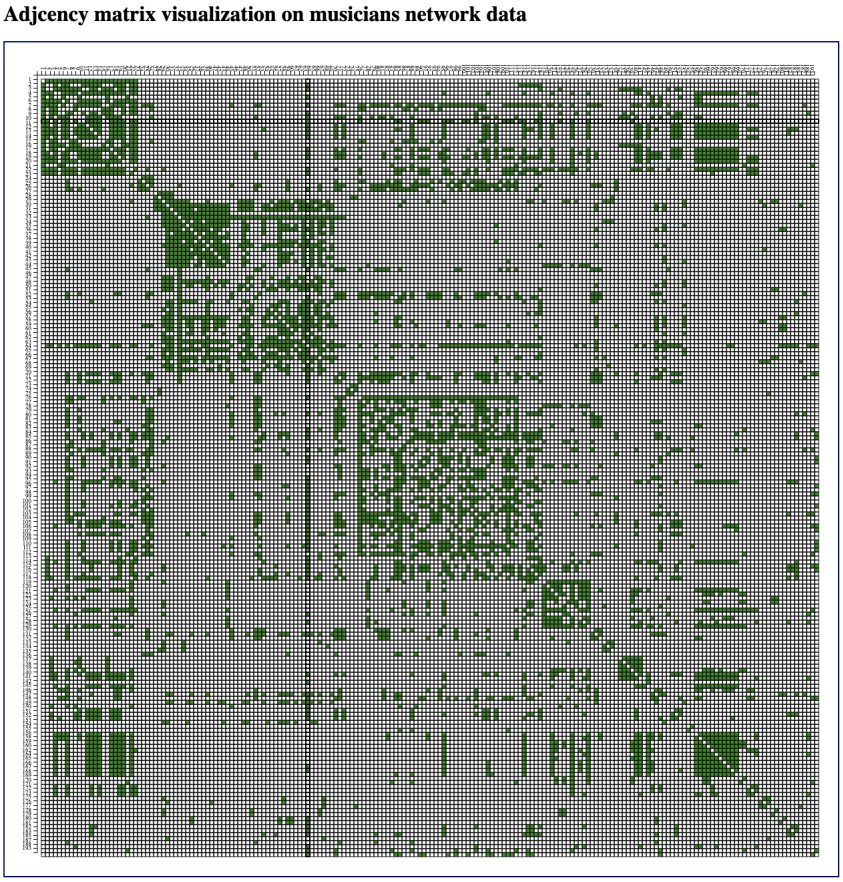


Figure 9: Adjacency Matrix Visualizations on Jazz Musicians Network

Q3. What are the user-interactions implemented for each visualization? Which, in your opinion, are user-interactions which enhance your visualization?

User Interaction implemented for each visualization is discussed in their respective sections. The common user interaction among all the visualizations is the hovering effect of the mouse. The impact of it varies based on the visualizations. But this helps the user to identify the entity he wants to see, and similar to it brushing technique is used in multivariate visualizations to select a specific set of elements, thereby preventing occlusion. We also allowed the user to select variables using the drop-down list in hierarchical visualization. All these interactions will enable the user to choose the entities he wants without restrictions are better user interactions.