CS 732: Data Visualization Assignment 2 Report

M Srinivasan
IMT2021058
IIIT Bangalore
m.srinivasan@iiitb.ac.in

Siddharth Kothari
IMT2021019
IIIT Bangalore
siddharth.kothari@iiitb.ac.in

Sankalp Kothari
IMT2021028
IIIT Bangalore
sankalp.kothari@iiitb.ac.in

***Abstract*—Assignment 2 report made by group "Are you able to visualize?" for the course CS-732 Data Visualization. The colour and contour mapping has been done on the INCOIS on the dataset "Surface Rain Rate"[1]. The node link diagram has been plotted on the Congress Votes dataset available on Konect [2]. The quiver plots have been plotted on the "OSCAT wind" dataset [31]. The treemap is done using the dataset from A1.**
This report also mentions all the observations made through various visualizations of the data.

I. Dataset

A. Motor Vehicle Collision Dataset-NYC

The given dataset is about all the motor vehicle accidents in NYC from the year 2020-2023 (July). Some of the important data the dataset contains is listed below:

- Date of collision
- Time of collision
- Borough of New York City in which the accident had taken place.
- The latitude and longitude of the location where the accident took place.
- Number of people injured and killed (pedestrians / motorists / cyclists)
- Major contributing reason to the accident etc.
- Street where the accident occurred in the borough.

A grouping similar to the one used for assignment 1 for the accident categories has been used for this assignment as well.

B. Surface Rain Dataset

The given dataset is about the sea surface rain rate (in $mm\ h^{-1}$). The data is taken for 9 dates (1,11,and 21 of May, Jun, Jul 2015).

The data is present as .txt file where each row contains the values for a single latitude, while each column is the data for a single longitude. Note that we only have data from 0 to 250° longitude (for some latitudes, we do not have all the data), and hence we have only plotted data for the corresponding longitudes.

In this dataset, -999 indicates a bad value, all the remaining values are valid.

C. Oscat Wind Dataset

The given dataset is about meridonal and zonal wind speeds across the globe. The data is present in a tabular format containing longitudes as columns and latitudes as rows. Each cell contains the direction and magnitude of wind velocity at that location. We have collected the data for 6 random days and have plotted them for our visualization. The 5 random days are 30 May, 30 Jun, 15 Jul, 30 Jul and 31 Aug, 2013.

We have used both the zonal and meridonal wind speed data to plot the streamline and quiver plot.

- Positive value indicates that the direction of wind is from West to East(Zonal) and from South to North(Meridonal).
- Negative value indicates that the direction of wind is from East to West(Zonal) and from North to South(Meridonal).

The data had a lot of garbage values *-1.e+34* in particular. We had replaced it by value zero for plotting of the quiver and streamline plots. Since the dataset was too large and ranged across the globe, the visualization for the enitre global map rendered nothing useful so we had decided to plot only for a smaller section of the globe. We had chosen the area near Indian subcontinent for plotting of quiver and streamline plots.

D. Congress Votes Dataset

This dataset contains a list of edges between various nodes. The nodes represent politicians speaking in the United States Congress, and a directed edge denotes that a speaker mentions another speaker. The weight of an edge (positive or negative) denotes whether the mention is in support of or opposition to the mentioned politician. Multiple parallel edges are possible. Loops are allowed, i.e., speakers may mention themselves.

II. Libraries

In Python, we have used *numpy, pandas* for handling data and *matplotlib, cartopy* libraries in Python to create the visualizations for this assignment.

We have also used *JavaScript* for browser based visualization plots (Treemaps and Parallel-Coordinates plot), and *Gephi*[7] for network visualization.

We have also used CSV to Json online converter for handling data[6].

III. Visualizations

A. Colour Mapping

We have used the surface rain rate dataset for the aforementioned dates for these plots. The bad values (-999) have been masked and excluded from the plot.

A limitation of the dataset which was available in ASCII format, is that the size of the line limited the number of longitudes for which we have accurate data. Hence we have a function to determine the longitude till which we have data,

across all the datasets, and we plot the longitudes till that point. We observed that this longitude is 249.375°. For the remaining longitudes, we first append the bad value (-999) wherever required, then plot and limit the extent of the map actually shown.

We use the matplotlib library for making the plots, along with PlateCarree projection of the cartopy library to add the map features such as the longitudes and latitudes, the coastline and land features on the map. For animations, we use the matplotlib.animation library. All the plots have been made on the dataset for 1 May 2015, and masking has been carried out for bad values.

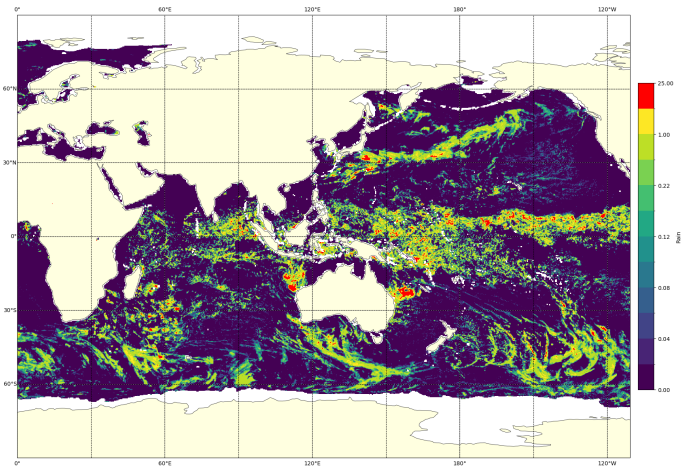

Fig. 1. Viridis Discrete Colormap

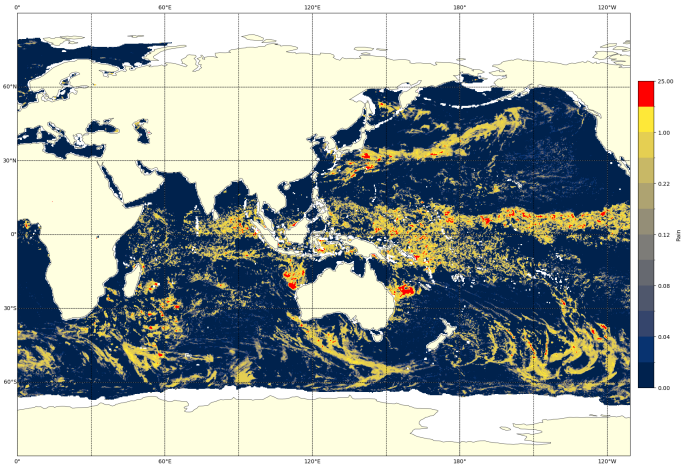

Fig. 2. Cividis Discrete Colormap

The procedure followed for this section is as follows -

1) We first preprocess the data as described above, and plot till the longitude 249.375°
2) We first count the number of values in predefined ranges of data. These ranges have been picked from the INCOIS website itself, and we find that they give aesthetically pleasing plots when used with discrete colormaps.

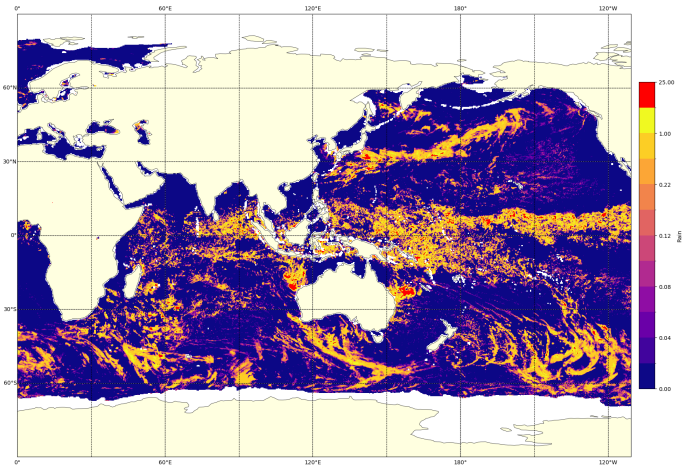

Fig. 3. Plasma Discrete Colormap

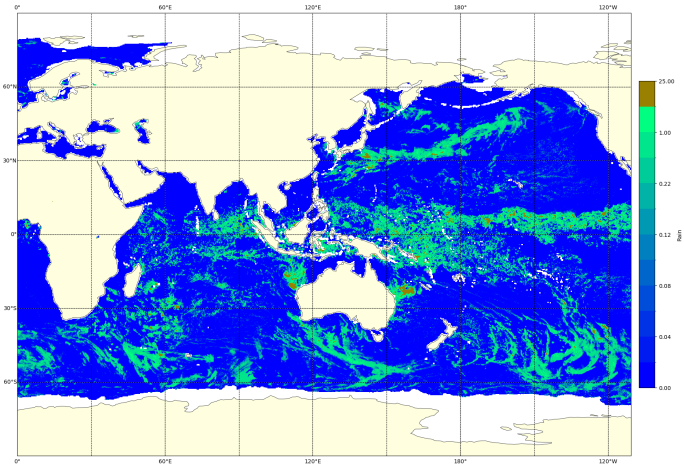

Fig. 4. Winter Discrete Colormap

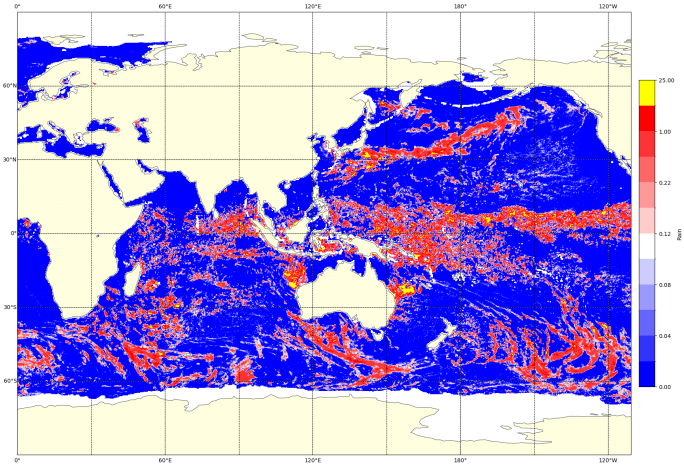

Fig. 5. Bwr Discrete Colormap

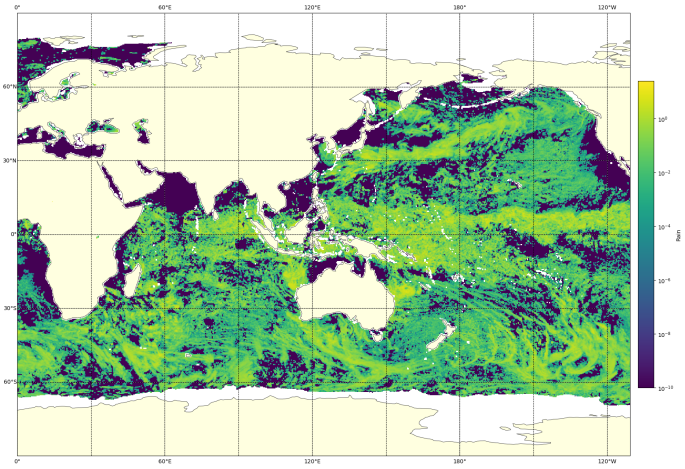

Fig. 6. Viridis Logarithmic Colormap

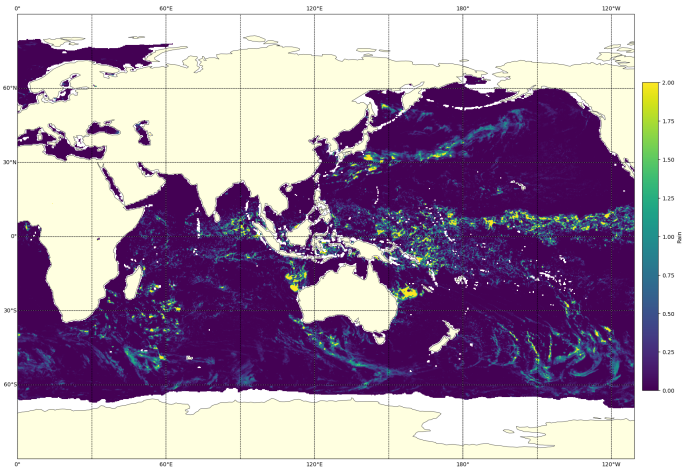

Fig. 7. Viridis Continuous Colormap

3) We then begin experimenting with the various scales. We plotted data for continuous, discrete and logarithmic scales, in the Viridis colormap.
4) For the continuous case, we keep the max value in the colormap as 2, any values beyond 2 take the same colour as 2. This is done as if we take the max value (which is as large as 25), then the entire map appears as the same colour, due to the fact that most of the values in the dataset are very close to 0, with very few outliers. For the logarithmic scale, we add a small offset to the good values (values not equal to -999). This is done to handle zeroes in the data, where the logarithm is not defined.
5) We observe that the plots are more easily understandable when we use the custom ranges from the INCOIS website, as compared to the continuous and logarithmic scales. For the discrete case, we observe that there are outliers in the data, which can be handled by adding an additional colour to the colorscale. Hence discrete colormap is best suited for our needs.
6) We now experiment with various colormaps. We start with the diverging colormap - bwr (blue white red), however we realise that since the data itself is sequential, the diverging colormap does not make much sense.
7) We then move on to sequential colormaps. The colormaps used are - viridis, cividis, plasma and winter. Winter does not give a good plot as it is not a gradual colormap, and is not suited for the task.
8) The other three, particularly viridis, lead to easily discernable and understandable plots, and we feel that this combination of discrete colormaps along with viridis/cividis/plasma colormaps are the best suited for this task.
9) We have also added animations for each of these cases discussed here.
10) We experimented with global and local maxima and minima, but we did not include those as most of the values are concentrated near zero, and hence the maximas were of no real consequence for plotting. Instead we went with the scale described above.

B. Contour Mapping

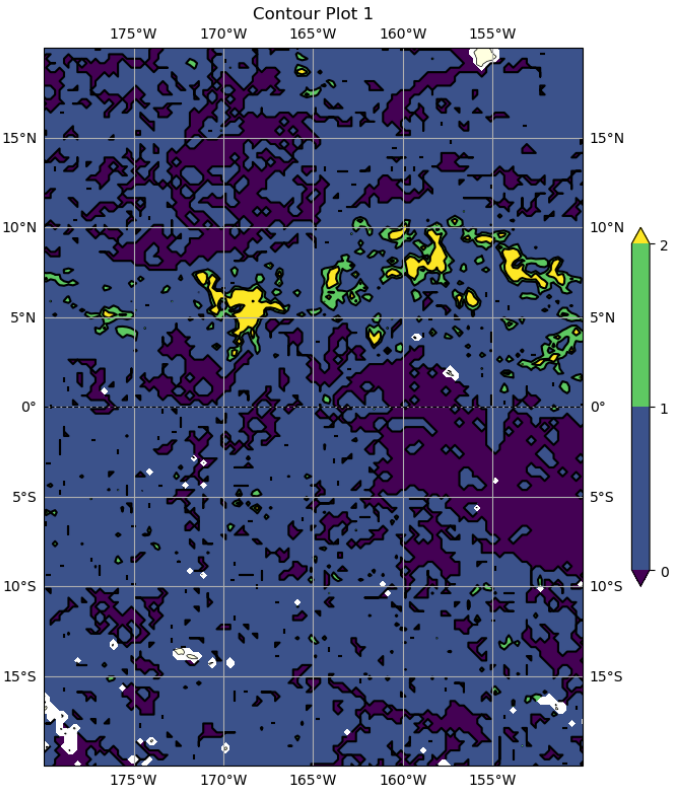

Fig. 8. Contour plot 1

We have used the surface rain dataset again for the aforementioned dates in these plots. The bad values (-999) have been masked and excluded from the plot.

The constraints remain the same as prior. We are using matplotlib for making the contour lines, and cartopy to add the map features, and an online tool to combine all the plotted contour maps into a video.

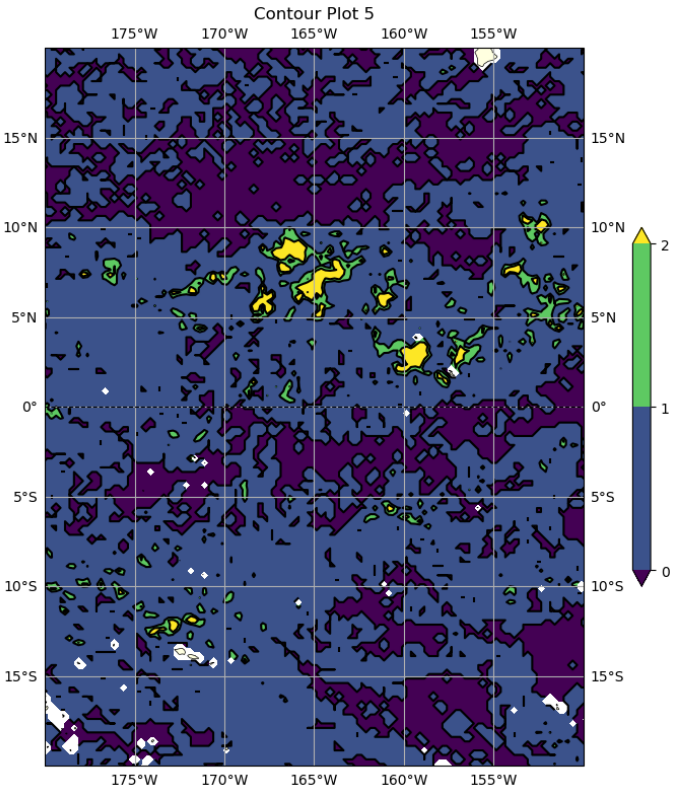

Fig. 9. Contour plot 5

Since the data was very closely plotted, we have taken a small cartographic region for the contour (150-180°W longitude and -30°S to 30°N latitude) plots so that the lines are clearly visible.

Figures 8 and 9 represent a couple of the contour plots. We have used the contour library function from matplotlib library which makes contour line plots.

The contour function internally implements a Marching Squares Algorithm to make the contour lines. We then filled the regions to make them more visible. The reason for using Marching Squares is to add a more visible distinction in the contour regions, and then we fill those regions to add a clearer understanding of the values represented by the contour regions.

C. Quiver and Streamline plots

The direction of arrow in the quiver plot is based on the meridonal and zonal wind speed for that location.

As mentioned earlier we have chosen 5 random days (Dates already mentioned)from the available dataset and then plotted the quiver and streamline plots for the same. There were lot of garbage values(-1.e+34) and we had to replace them by zero for plotting of the quiver and streamline plots. Matplotlib was used for plotting the plots and for creating the required the giphs from it.

Figure 10 and Figure 11 represent the quiver and streamline plot respectively for the same date.

We have used the *PlateCarree* projection for plotting the arrows on the map.

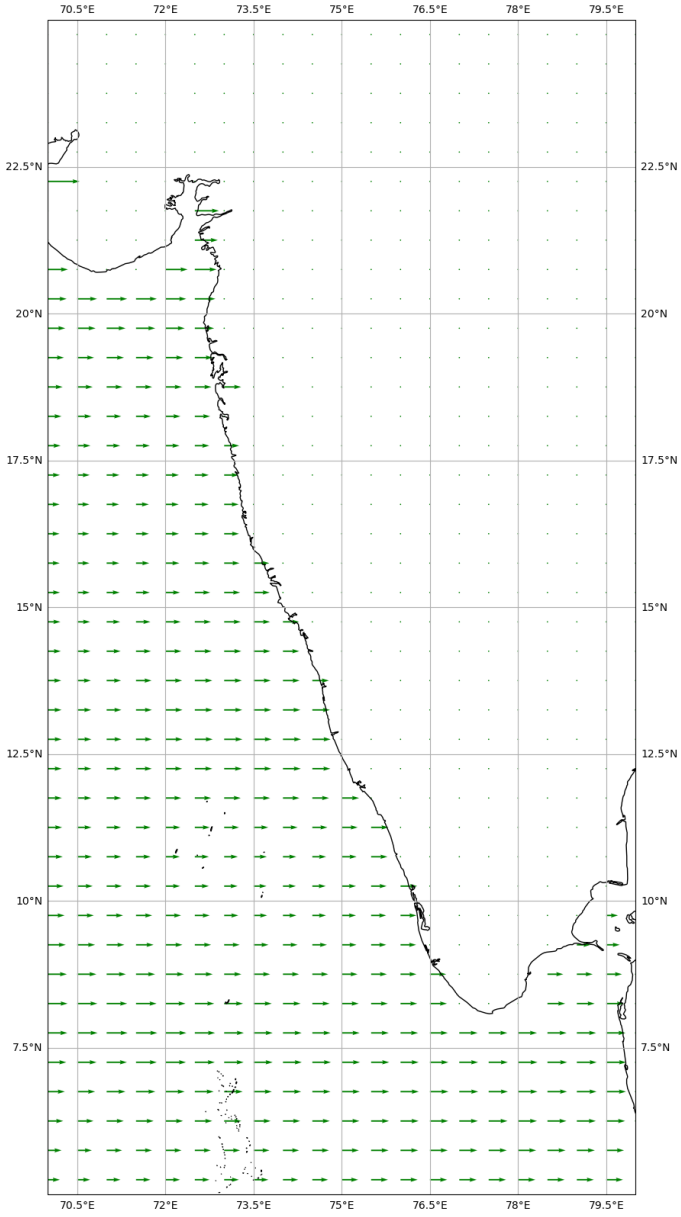

Fig. 10. Quiver Plot for 31-May-2013

D. Node Link Diagrams

The node link diagrams have been made using Gephi, in the following manner - we first plotted the positive and negative edges with weights 1 and -1 in the same graph. The software does not support multiple parallel edges, which necessitated the use of weights, but it leads to an issue - if person A talks 3 times positively about a person B, and once negatively, the overall weight becomes 2. However we intended to show the person talking positively 3 times and once negatively, as compared to the aggregate.

To take care of this, we first processed the data to convert it into a format which can be used by Gephi. For this, we first make a nodes.csv file which contains the nodes along with the labels, following which we create edges_positive.csv

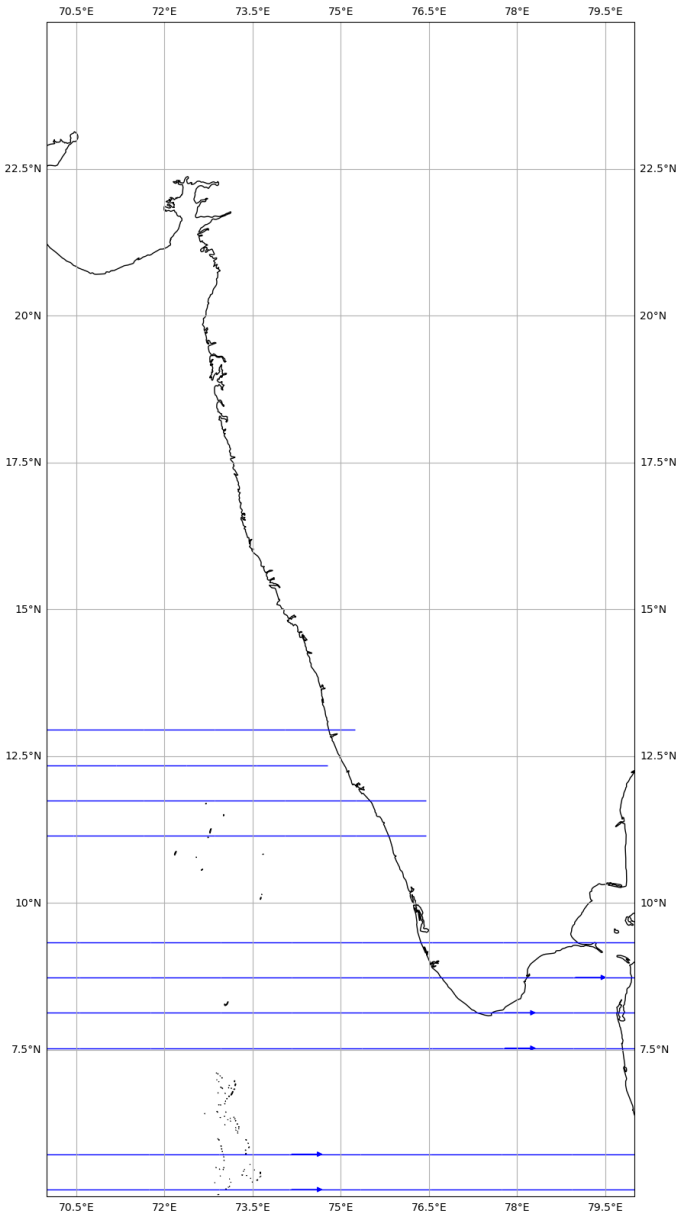

Fig. 11. Streamline Plot for 31-May-2013

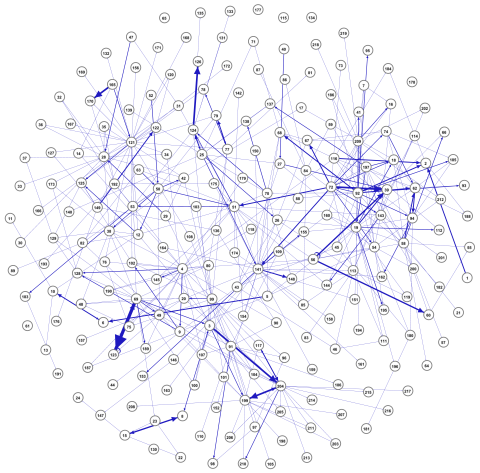

Fig. 12. Node link diagram with positive edges and Fruchterman Reingold layout

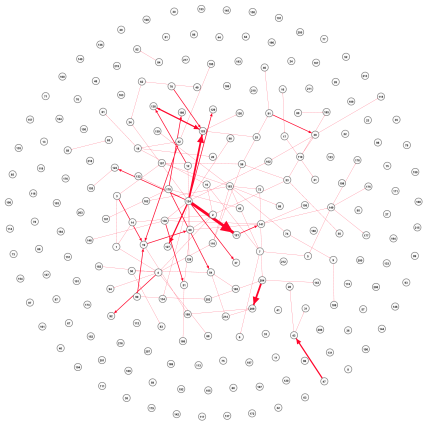

Fig. 13. Node link diagram with negative edges and Fruchterman Reingold layout

and edges_negative.csv files for the positive and negative edge plotting.

Following this, we plotted the positive and negative edges separately, following the same convention for colour. The thickness of the edge now indicates the number of times a person spoke postively (or negatively) about another person. We experimented with three graph layout algorithms - namely OpenOrd, Fruchterman-Reingold and Yifan Hu. The plots have been included in the report. (Figure 12-17)

E. Treemap Plots

We added 2 treemaps of depth 2 for the dataset that we had received for A1. We did a bit of preprocessing on the data and wrote the data into a csv file, as labels, their parents and values. All the labels having the same parent, have the same color in the treemap.

The dataset from A1 corresponds to Motor Accidents in NYC boroughs and the reason for those accidents.

For the first treemap, similar to A1, we divided the data into 7 broad categories and added a separate column in the data for the same, as is shown in the file **accident_type.ipynb**. Then we took the primary cause of the accident as the label and the broader category as the parent for the treemap. The figures 18, 19 show two different variations of the tree plot - squarified and snake plots. We can see that the data is better seen through the squarify plot, for this large number of labels where the difference in values is large.

For the second treemap, we took the parent as the borough and the street on which the accident occurred as the label for the treemap. The preprocessing can be seen in

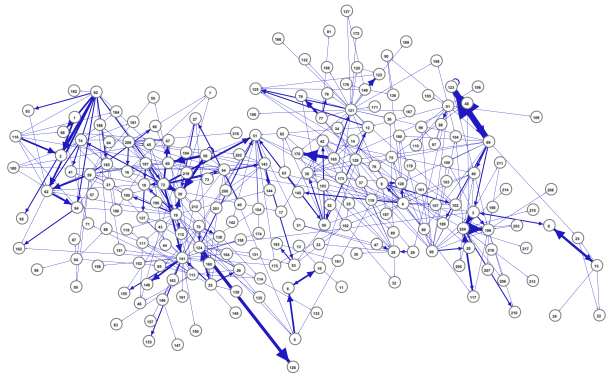

Fig. 14. Node link diagram with positive edges and OpenOrd layout

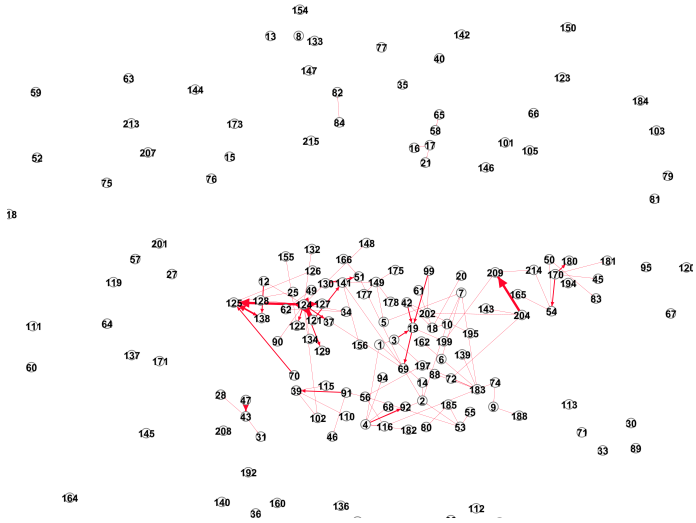

Fig. 15. Node link diagram with negative edges and OpenOrd layout

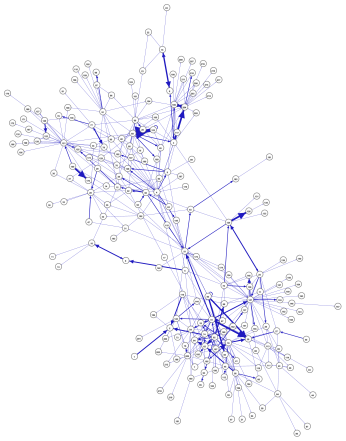

Fig. 16. Node link diagram with positive edges and Yifan Hu layout

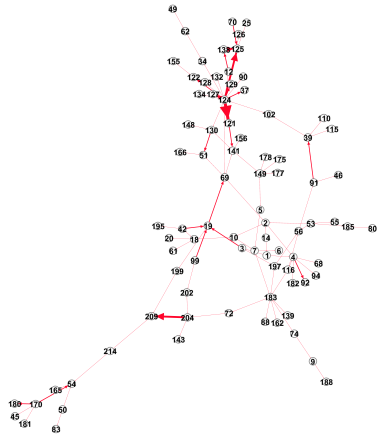

Fig. 17. Node link diagram with negative edges and Yifan Hu Layout

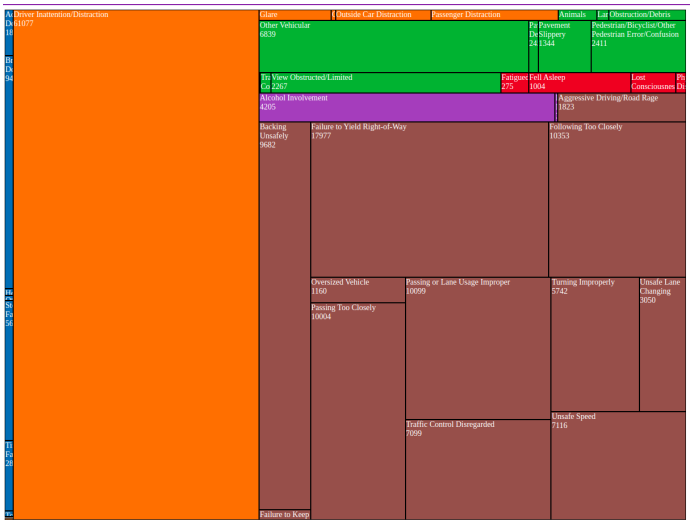

Fig. 18. Squarify Accident Cause plot

borough_street.ipynb. The figure 20 shows the treemap for the same.

F. Parallel Co-ordinates Plot

We have used our pre-existing dataset for plotting the parallel co-ordinates plot. We had a lot of categorical data in our dataset and we had to remove most of them.

We have removed the rows which contained null values in the columns that we were plotting as axis in the parallel co-ordinates plot. We have basically plotted the parallel co-ordinates plot for two kinds of inferences -

1) We had plotted to few accident types to analyze the time of day for those kind of accidents.
2) We have also plotted them for various kinds of accidents to see the leading cause of accidents.

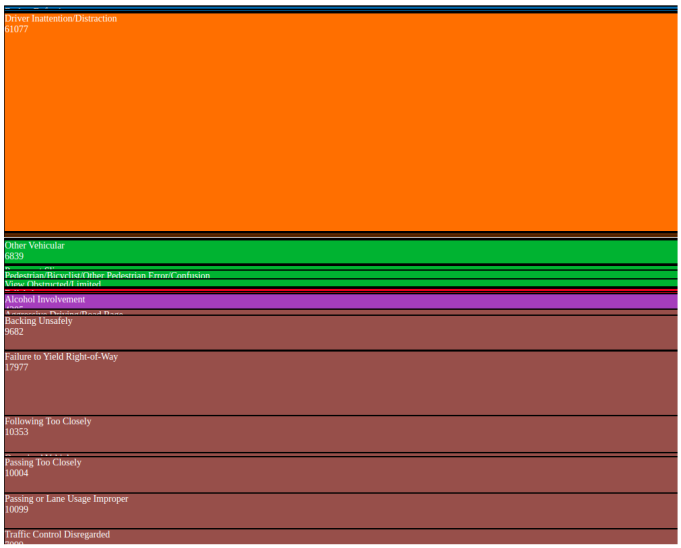

Fig. 19. Snake Accident Type plot

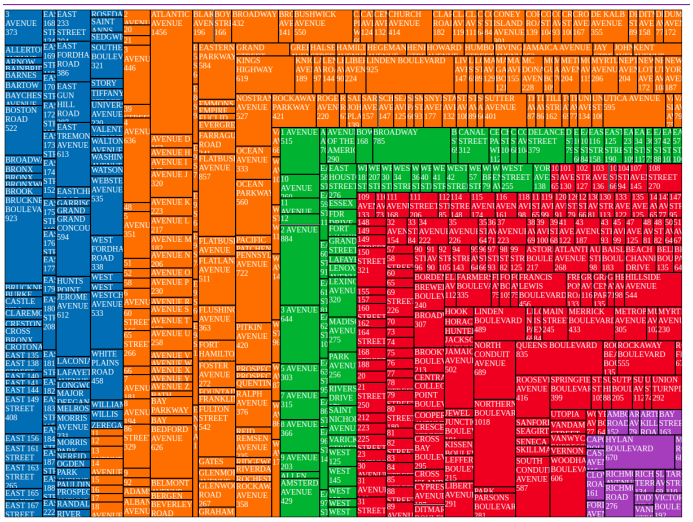

Fig. 20. Squarify Borough-Street plot

We have plotted the parallel co-ordinates plot for two accident types namely: 1) Medical Reason/Fatigue and 2) Substance/Alcohol Abuse.

The third plot we drew was based on the kind of accidents that are taking place in the city.

We had divided our data into four parts based on the time of day analogy as mentioned in the first assignment.

We had allocated the following colour schemes for the parallel co-ordinates plot(Only for Figure 21 and Figure 22) as follows:

- Midnight - Orange
- Morning - Green
- Afternoon - Blue
- Night - Red

But since many of the accident types had similar kind of distribution over the axis. Many had overlapping lines and the

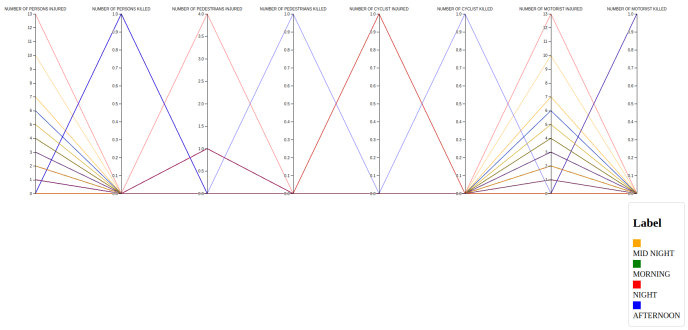

Fig. 21. Parallel Co-ordinate plot for Medical Accidents

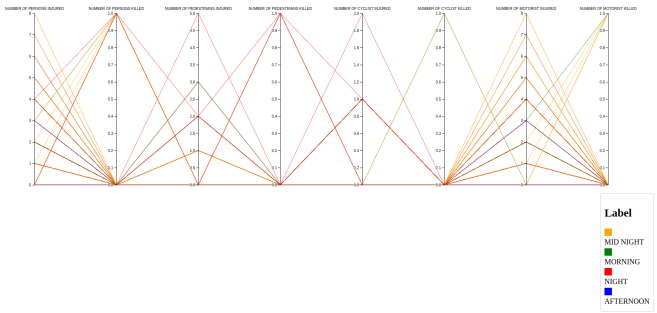

Fig. 22. Parallel Co-ordinate plot for Alcohol/Substance abuse Accidents

colours started to mix after a certain point.

The plots have been interactive to the users. Axes in all of the plots can be moved to make the necessary visualization with it's neighbours. User can also see a subset of data of their interest by brushing. The lines corresponding to a data point will be highlighted by clicking and dragging on the axis.

The axes names can't be read clearly on the report so we have not included the picture of the shifted axes parallel co-ordinates plot here. It has been added to the images folder.

IV. Inferences

1) The color maps and contour maps pertaining to surface rainfall data indicate the regions of low vs high rainfall.

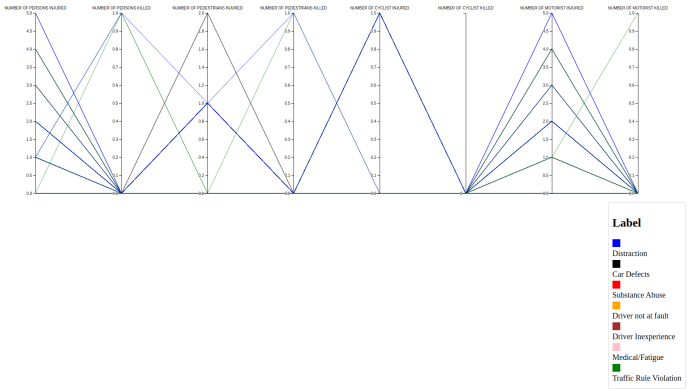

Fig. 23. Parallel Co-ordinate plot for various kinds of accidents.

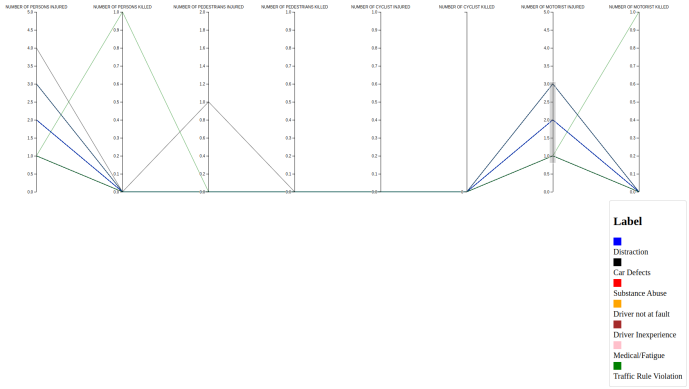

Fig. 24. Brushing Effect on the parallel co-ordinates plot for accident types

2) Based on the animations generated by the colormap, we observe that the coastline of India has a gradual increase in the rainfall amount as we progress from May to June. In July, there is a decrease which is probably due to the fact that the rain has moved to the Indian Subcontinent rather than being concentrated at the coastlines. At the end of July, we observe that the rainfall at the coastlines increases again, indicating the end of the monsoon season, and the gradual movement of the rains out of the subcontinent.
3) This pattern is consistent with the actual pattern of winds that we observe around the monsoon season.
4) The Pacific Ocean has a very disproportionately high amount of rains, throughout the period observed for the visualizations, making it the deadliest of the five oceans.
5) The color maps also make it evident that the rainfall near coasts of islands (especially near the Indonesian and Australian coasts) is higher than in regions which are not near coastlines.
6) The contour plots not only give the regions of varying rainfall, but the also help to differentiate the amount of rainfall across the different dates for a smaller area.
7) The variation in both can be used to study the amount of rainfall across the season.
8) The node link diagrams show that speakers in the United Nations mostly speak positively about other speakers, as can be seen in the scarcity of negative edges as compared to positive edges.
9) Regarding the layouts, we feel that Fruchterman Reingold gave the most aesthetically pleasing graph. Other layouts caused issues such as overlapping of nodes, which we had to resolve manually, and some nodes were very far away from others. In Fruchterman Reingold, even disconnected nodes were close enough for us to make required inferences easily.
10) The treemap for the accident type shows that there are 2 broader accident categories with smaller subcategories, which constitute the cause for most accidents in NYC.
11) The treemap for the borough and street has more evenly sized blocks, indicating that accidents can happen anywhere. Although, the sizes for the borough blocks itself, is larger for Brooklyn and Manhattan, indicating they are the 2 busiest boroughs in NYC and sometimes highly unsafe in terms of road accidents.
12) We have plotted the wind speed direction along the Arabian coast for the Indian Sub-continent during the periods of South-west monsoon.
13) As expected we can see that the winds during the month of May flow parallel to the Tropic of Cancer.
14) The winds start hitting the West coast along the North-East direction as expected in June. This is the onset of the South-West Monsoon. During the months of June and July the winds continue to hit the the west coast and the speed, direction of winds are greatly varying. Towards the month of August, we see the winds are receding back and this marks the end of the South-West monsoon.
15) In the parallel co-ordinates plot for the Medical accidents type, We see that there are lot of thick red and yellow lines which indicated that most of the accidents are taking place during the night and mid-night hours.
16) In the parallel co-ordinates plot for the Alcohol/Substance abuse accidents, as expected, we can see that there are lot of thick red and yellow lines which means that most of the accidents are taking place during the night and mid-night hours.
17) In the parallel co-ordinates plot for various kinds of accidents taking place in the New York City. We can see a lot of blue and green lines more than the other ones which indicates that there are more Traffic Rule violation and Distraction related accidents.

V. Work Distribution

1) The color mapping and the node link diagrams were done by Siddharth Kothari (IMT2021019).
2) The contour mapping and the treemaps were done by Sankalp Kothari (IMT2021028).
3) Due to the data being the same for colour and contour mapping, the preprocessing of the data for the SciVis task has been done jointly by Siddharth and Sankalp.
4) The parallel coordinates plot along with the streamlines and quiver plot and the pre-processing for the same was done by Srinivasan M (IMT2021058).

References

[1] Indian National Centre for Ocean Information Services (INCOIS) - Surface Rain Rate [Online] Available: https://las.incois.gov.in/las/UI.html
[2] Konect - Congress Votes [Online] Available: http://konect.cc/networks/convote/
[3] Indian National Centre for Ocean Information Services (INCOIS) - Oscat Wind Dataset [Online] Available: https://las.incois.gov.in/las/UI.html
[4] US data.gov catalog - City of New York - Motor Vehicle Collisions-Crashes [Online] Available: https://catalog.data.gov/dataset/motor-vehicle-collisions-crashes
[5] Matplotlib Documentation https://matplotlib.org/stable/index.html
[6] ConvertCSV - Website https://www.convertcsv.com/csv-to-json.htm
[7] Gephi Documentation https://docs.gephi.org/
[8] Cartopy Documentation https://scitools.org.uk/cartopy/docs/latest/
[9] d3.js documentation https://d3js.org/d3-hierarchy