CS732/DS732: Data Visualization – Datathon 1

Kunika Valecha

September 05, 2020

1 Problem Statement

- To generate scalar field visualizations on Ocean dataset for scalar quantities viz, sea surface salinity (SSS), sea surface temperature (SST), sea surface height anomaly (SSHA) using color mapping, contour plotting and elevation mapping schemes.
- To generate vector field visualizations on the same dataset for vector quantities viz, zonal and meriodional current values using quiver plotting scheme.

1.1 Data Description

The data source is Ocen Dataset under "Ocean Analysis" and "5-day ice". 5-day moving average of values computed of Indian Ocean from the ocean model MOM, run by the Indian National Center for Ocean Information Services, INCOIS, Hyderabad.

- **Metadata**: The dataset consists of 5 folders, each for different variables, namely, sea surface salinity (SSS), sea surface temperature (SST), sea surface height anomaly (SSHA), meridional current, and zonal current, during the period December 2003 – December 2005. The datasets are at 5-day interval, thus giving 147 timesteps. Each of the folders also has a screenshot of the webpage giving latitude-longitude ranges of each variable.
- The datasets for SSS, SST, and SSHA have data for 187 longitudes ranging in [29.8892W,119.8237W], and 188 latitudes ranging in [29.7511S, 29.7511N]. Thus, each data file for a timestamp has 187*188=35156 scalar values.
- The zonal and meriodional current values can be treated as a single 2-dimensional vector field,where, zonal current speed (east-to-west speed, along latitude) can be assumed to be u- (or x-) component, and meridional current speed (north-to-south speed, along longitude), as v- (or y-)

component. The datasets for current values have data for 181 longitudes ranging in [30W, 120W], and 189 latitudes ranging in [30.0005S, 30.0005N]. Thus, each data file for a time-stamp has 181*189=37209 scalar values in each file.

1.2 Data Reading and Cleaning

- Each scalar quantity has a folder of files. And each file contatins data tabled into a matrix of [35156x5]. Where the columns hold the quantities- *DATETIME, TIME, LONGITUDE, LATITUDE and SCALAR QUANTITY.*
- The faulty readings have been given a BAD FLAG : -1.E+34.
- Before plotting the visual fields (either scalar or vector). It is necessary to remove the rows containing these values for any of the columns. For this reason we have eliminated all rows from dataset which contained the BAD FLAG.
- We can also choose to drop only the rows for faulty LONGITUDE, LATITUDE and TIME VALUES. Keeping, the faulty scalar values, and representing them with some colour which represents one of the extremities of our colorbars. This is an approchable idea if the elimination of Faulty observations making your data very sparse. However, in this particular dataset this is not the case, hence we have choosen to drop the faulty reads.

2 Visualization of scalar fields

Scalar field visualization is the graphical expression of relationships between scalar values distributed in space. The difficulty of rendering scalar fields depends on a number of factors; The dimensionality of the data, whether the data has been sampled on a regular grid or not, and the range of values the field can assume.

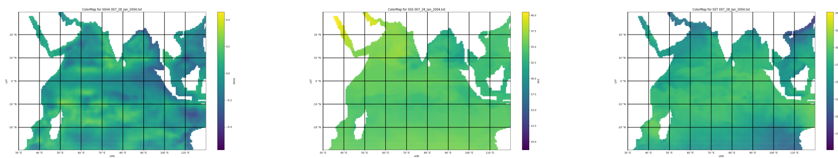
There exists multiple representations of scalar data. The commonly used in practised are enlisted below with examplar plots on the Ocean Data.

2.1 Color Maps

A color map (often called a color table or a palette) is an array of colors used to map pixel data (represented as indexes into the color table) to the actual color values.

- The range of scalar quantity in our data is corresponded with a range of color gradient.

- The Latitude and Longitude coordinates are aliased as x and y coordinated respectively. The *colormaps* for different quantities can be visualised as below,



(a) SSHA (b) SSS (c) SST

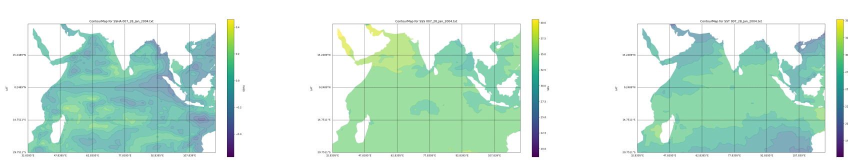
Figure 1: Colormaps for different scalar quantities in ocean data

2.2 Contour Maps

Contour maps are a way to depict functions with a two-dimensional input and a one-dimensional output. With graphs, the way to associate the input *(x,y)* with the output *f(x,y)* is to combine both into a triplet *(x, y, f(x, y))* and plot that triplet as a point in three-dimensional space. The graph itself consists of all possible three-dimensional points of the form *(x, y, f(x, y))* which collectively form a surface of some kind. But sometimes rendering a three-dimensional image can be clunky, or difficult to do by hand on the fly. Contour maps give a way to represent the function while only drawing on the two-dimensional input space.

The *contour maps* for different quantities can be visualised as below,

- The range of scalar quantity in our data is corresponded with a range of color gradient.
- The Latitude and Longitude coordinates are aliased as x and y coordinated respectively. The contour maps for different quantities can be visualised as below,

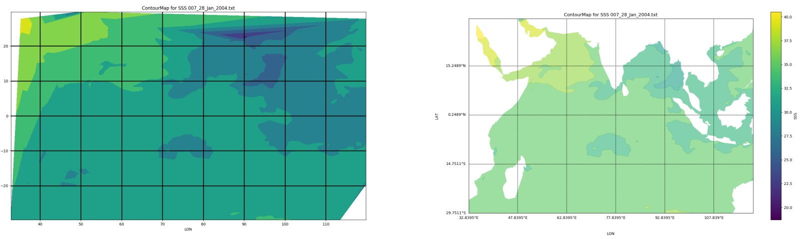


(a) SSHA (b) SSS (c) SST

Figure 2: Contour maps for different scalar quantities in ocean data

2.2.1 Masking in Contour Plotting

Masking is done to clarify the legibility of a map that's packed gradient and contour lines. The contour maps with gradients give us very fine insights of the scale of change in the scalar quantity but at if not masked the gradient takes the non-relevant parts into the account and hence the range of spread of gradient increases giving us difficulty in observing the changes ans well as regions. An example of same image with and without contour can been seen in figure below,



(a) contours without masking

(b) contours with masking

Figure 3: Contour maps for same image with and without masking

2.3 Elevation Maps

Elevation Mapping gives us the visualization in 3D plot which shows the variation in the scalar field across the longitude and latitude. For rendering the data on a 3D plot, Matplotlib's plotsurface function enabled us to do the same.

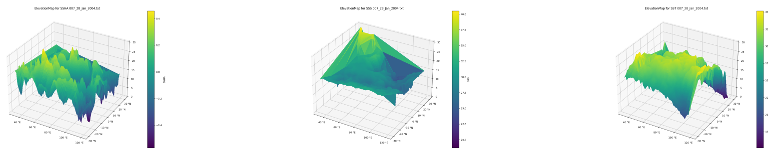
Before plotting we need to normalize the value of scalar quantity so that it becomes scalable on z- axis.

The scalar value when plotted on an elevation map will display the peak and the valley corresponding to the scalar data. The highest point is much more prominently visible in the visualization. To make the lowest point equally prominent we performed normalization as in the color map giving positive values to pressure data and negative values as the pressure in ocean showing peak drop in pressure and sheer drop in temperature at hurricane eye in terms of elevation.

The *Elevation Maps* for different quantities can be visualised as below,

- The range of scalar quantity in our data is corresponded with a range of color gradient and normalized to scale on z-axis.

- The Latitude and Longitude coordinates are aliased as x and y coordinated respectively. The contour maps for different quantities can be visualised as below,



(a) SSHA (b) SSS (c) SST

Figure 4: Elevation maps for different scalar quantities in ocean data

3 Visualization of vector fields

Vector fields let you visualize a function with a two-dimensional input and a two-dimensional output. You end up with, well, a field of vectors sitting at various points in two-dimensional space.

You can visualize a vector field by plotting vectors on a regular grid, by plotting a selection of streamlines, or by using a gradient color scheme to illustrate vector and streamline densities. You can also plot a vector field from a list of vectors as opposed to a mapping.

There exists multiple representations of vector data. The commonly used in practised are enlisted below with examplar plots on the Ocean Data.

3.1 Quiver Plot

A quiver plot is a type of 2D plot that shows vector lines as arrows defined by the components u and v at points described by the coordinates defined by x and y of dimension. The component U is zonal current (m/sec) and component V is meridional current (m/sec), according to the given dataset.

The *Quiver Plot* for different quantities can be visualised as below,

- A quiver plot displays velocity vectors as arrows with components (u,v) at the points (x,y).For example, the first vector is defined by components u(1),v(1) and is displayed at the point x(1),y(1).

- quiver(x,y,u,v) plots vectors as arrows at the coordinates specified in each corresponding pair of elements in x and y. The matrices x, y, u, and v must all be the same size and contain corresponding position and velocity components.

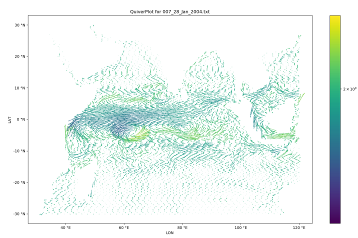
- quiver(u,v) draws vectors specified by u and v at equally spaced points in the x-y plane.

- The Latitude and Longitude coordinates are aliased as x and y coordinated respectively.

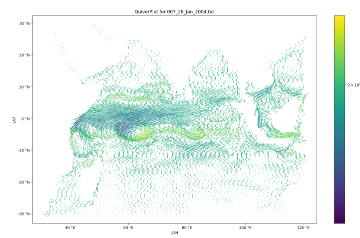
3.1.1 Making Quiver Plots deducible

For a large dataset, it becomes difficult to infer through the quiver plots.Because high density of the arrow heads make their size smaller(which is generally regulated by automatic arrow scaling). This makes the job analysing the arrow heads, nearly impossible. Solution to this problem is sparsing of the dataset.In order to make the dataset sparse, the every n^{th}. row can be dropped from dataset. The value of n depends on the size of data set.

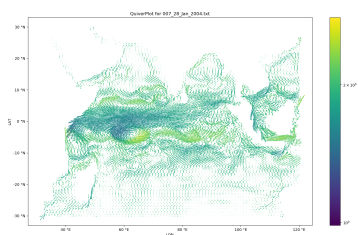
The *quiver* maps for different values of n can be visualised as below,



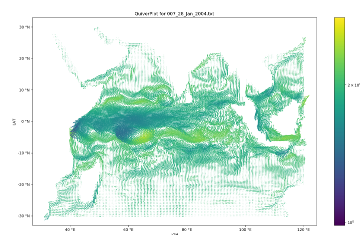
(a) n=4



(b) n=3



(c) n=2

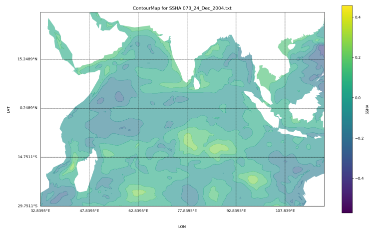


(d) n=1

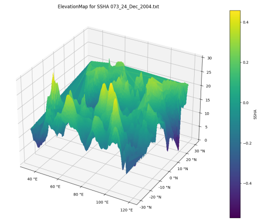
Figure 5: Quiver plots for different rate of sparsing in ocean data

4 Effects on variables of Indian Ocean due to *Tsunami* on December 29, 2004

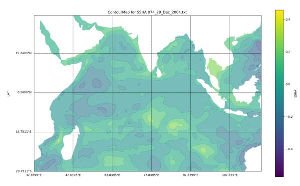
- ***Sea Surface Height Anomaly*** Not a very remarkable change in Height Anomaly is observed. However, a relative depletion can be seen on across latitude 3° 19' N and longitude 96°E. These coordinates are near to the epicentre of the *Tsunami*. The depletion continued to persist till early 2005. And also the values did not hit the bottom most of all. Hence, it is not likeable to be a cause of *Tsunami*. A good insight of these observations can be taken from *figure 6*.



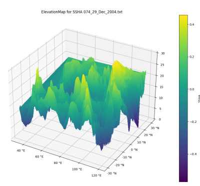
(a)



(b)



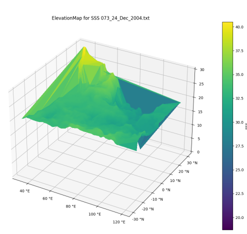
(c)



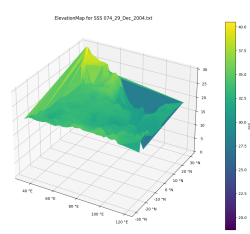
(d)

Figure 6: fig(a), (b), (c), (d) showing contour plot and their corresponding Elevation maps recorded on December 24, 2004 and December 29, 2004 respectively. Contour maps highlight the areas where the change in scalar is observed and elevation mapping provides with clear picture of intensity of the change.

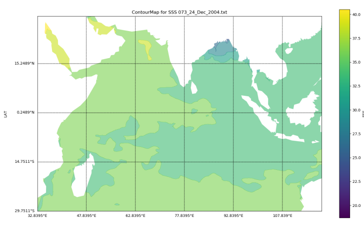
- ***Sea Surface Salinity*** A very slight raise in salinity has been encountered near Sri Lanka and the southern belt of India post *Tsunami* which trailed till early January, 2005 and then dropped more than the levels prior to *Tsunami* . A very similar pattern can be observed in BAy of Bengal and Nepal belts as well. A good insight of these observations can be taken from *figure 7*.



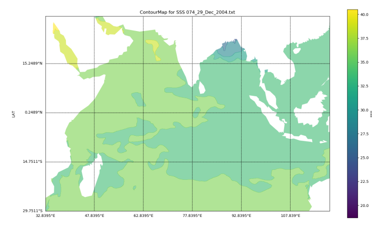
(a)



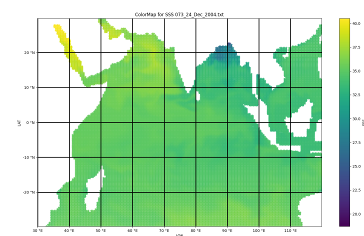
(b)



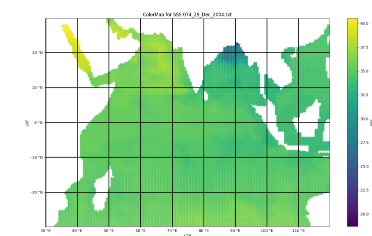
(c)



(d)



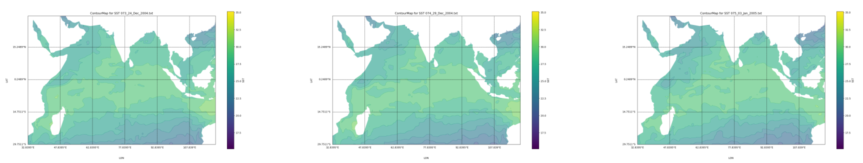
(e)



(f)

Figure 7: fig(a), (b), (c), (d), (e), (f) showing records on December 24, 2004 and December 29, 2004 respectively. Figure depict that contour plot and elevation maps fails to capture the very slight change of salinity which is visible in color maps.

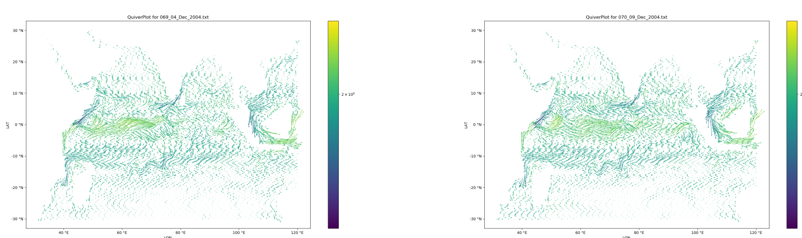
- ***Sea Surface Temperature*** On coordinates around latitude 22° 30' N and longitude 90° 5'E, near Nepal Belt, a fall in temperature is of around 2°-3° is found. At latitude 9° 83' N and longitude 101° 44'E near Gulf of Thailand, a very similar drop is observed. , Both of these looks to get spread across northern west side i.e. in the direction of ocean currents. A good insight of these observations can be taken from *figure 8*.



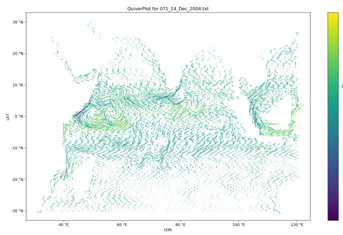
(a) (b) (c)

Figure 8: fig(a), (b), (c) showing contour plot and their corresponding Elevation maps recorded on December 24, 2004 ; December 29, 2004 and January 03, 2005 respectively. The spread in accordance with ocean currents can be referenced from *figure 9*

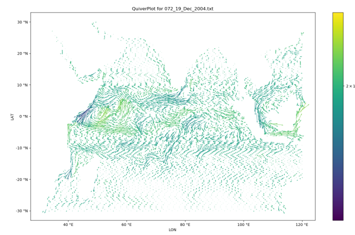
- ***Ocean currents*** Ocean currents are found to be quite denser in period of December 19, 2004 to December 29, 2004 in region latitude 0°N and longitude 80°E.A strong flow of currents can be observed from the 3° 19' N and longitude 96°E towards the West Northern side post the Disaster.A good insight of these observations can be taken from *figure 9*.



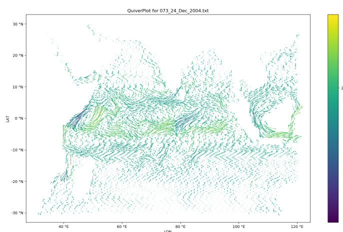
(a) (b)



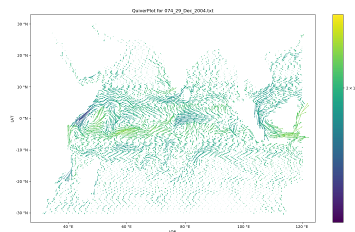
(c)



(d)



(e)



(f)

Figure 9: fig(a), (b), (c), (d), (e), (f) showing increase in ocean currents daywise

5 Inferences from the Data set

- temperature records show a rapid, continuous warming in the Indian Ocean, at about 1.2 °C (34.2°F).
- South of the Equator (20-5°S), the Indian Ocean is gaining heat from June to October, during the austral winter, while it is losing heat from November to March, during the austral summer. However, during the mid and end of year i.e. April and December respectively the Temperature on Northern most and southern most ends of Indian oceans looks very similar in the magnitude.
- North Western part of Indian Ocean is usually high in salinity and North Eastern part specifically Nepal belt is mostly low in the same. How ever other regions generally maintain a moderate level of saline nature.
- Height Anomaly is equally distribute all over the region throughout the times.

6 Libraries and Functions Used

6.1 Libraries Used

- Pandas for data loading, cleaning, clipping etc.

- numpy for some mathematical operations
- Matplotlib for Plotting the graphs
- sys and os for scripting
- Matplotlib Toolkits for plotting 3D surfaces and generating masks on contour maps.

6.2 Some important functions used

- **matplotlib.pyplot.scatter** *for plotting the colormaps.*
- **matplotlib.pyplot.tricontourf** *for plotting the contor maps.*
- **matplotlib.pyplot.plot_trisurf** *for plotting the elevation maps.*
- **matplotlib.pyplot.quiver** *for plotting the quiver plots.*
- **Basemap()** *for generating masks on continents during contouring.*