Disease Data Correlator

Shama Rashid

University of British Columbia

Vancouver, BC, Canada, V6T 1Z4

shama@cs.ubc.cal

ABSTRACT

In this paper I shall discuss the web based tool I developed to facilitate exploration of data related to particular disease prevalence among population of different regions in order to find data trends and to figure out correlation among different data attributes. I shall be using the data available from the website of Centre for Disease Control (CDC). I shall first mention some of the related works in the field and then I shall discuss my own implementation design. A direction for future extension of the software shall be provided later on in the paper.

KEYWORDS: Choropleth map, KML, infovis

1 INTRODUCTION

Disease prevalence data is a domain that is of interest to all people in general. Although the complexity of the visual representation of such data may vary depending on the goal of the viewer, the basic functionality required of all such representations is the geo-spatial distribution of the data i.e. the prevalence rate of a particular disease at different regions of the world or country. To understand why this sort of representation may appeal to a general user with no technical expertise in the visualization field or health science, you can consider the recent spread of H1N1 flu or even the seasonal flu. As a part of public awareness program, governments use this type of graphical representations to make the general population better comprehend the geospatial trend of spread and rate of population infected with that strain of flu. Having a map-based user-friendly representation may help a person to take appropriate measures to avoid travelling high-risk areas or to take preventive measures.

From the point of view of people related to health science and health engineering or even people responsible for policy making a more complex and interactive visualization is required to analyze trends and to explore further in search of trends. Based on these patterns recognized an effective policy can be formulated to ensure a better future for the people in terms of better health plans or area-wise customized plans.

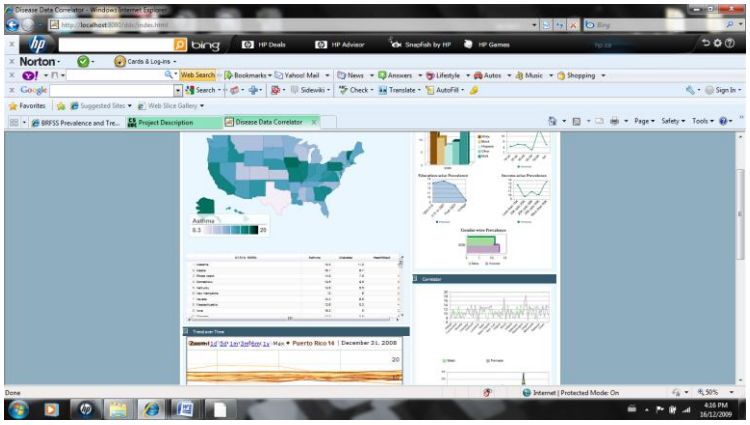


Figure A : The UI for the Disease Data Correlator aplication

2 RELATED WORKS

Lots of toolkits are available for thematic mapping and geospatial data representation. GeoServer, Google Thematic API, OpenLayers all deal with representing multi-layered maps. For particularly disease related data geospatial mapping there is a tool named Geoviz which is based on the GeoVISTA Studio developed by the department of Geography of Pennsylvania State University. This project started as an initiative from CDC to develop a user-friendly interactive tool for exploring disease data. However, I am uncertain about the current state of the project. GeoTools is an open source Java based library than can be used to implement basic map display related features from any Java application.

3 DISEASE DATA CORRELATOR

The solution I have implemented for the disease data correlator is based on the open source Google Visualization API. My solution focuses on representing the prevalence data encoded as colors for different regions. I have particularly displayed state wise prevalence data for the states of USA. I have also tried to add some of the analytical and exploratory tasks more relevant to people working in the health science field.

4 IMPLEMENTATION

I started out the project using GeoVISTA Studio which is an open software development environment designed for geospatial data. GeoVISTA is a programming-free environment that allows users to quickly build applications for geocomputation and geographic visualization. However, due to the unavailability of relevant

documentation or developer's guide, I began to search for an alternate tool or API midway along the implementation of the project. A few other options that I considered before settling on my final API choice are Geotools library for Java, Geoviz and GeoServer. All of these options were discarded due to the unavailability of proper guidelines for development or dependency on external libraries or modules.

My project is based on different packages of the Google Visualization API. I have used the Javascript version as opposed to the gadget version of the APIs. The API provides different packages for visualization of different types of data. The Choropleth map, which is the main component of the application, is drawn using the GeoMap package. A number of chart packages were used to display the demographic data representation and Visual Query Language was used to fetch relevant data from a Google spreadsheet.

The data available from CDC, my data source, was in a non-statistical format. To extract data and format it to the appropriate representation I developed a Java application.

I used DHTML, AJAX and Javascript snippets to develop the web based UI. The webpage developed was deployed using Apache Tomcat 6.0.2. Since some of the Google Visualization API events cannot be triggered if the page is accessed as local file, this was a necessary measure to access full functionality of the UI.

Following is a tabular summarization of the packages used from the Visualization API:

SL#	Package Name	Application
1	GeoMap	Map View
2	Table	Map View, Trend View
3	AnnotatedTimeSeries	Trend View
4	Area Chart	Details View
5	Bar Chart	Details View
6	Column Chart	Details View
7	Line Chart	Details View, Correlator
8	Visual Query Language	Data fetching from spreadsheet

I have used Javascript for formatting the data fetched using Visual Query Language from a Google spreadsheet available online to make it suitable as an input to the object instances of different packages. Also, Javascript was used to determine the actual query used in this task based on the settings of the variable pickers. For handling event triggers, Javascript was used once more to synchronize the views and taking appropriate actions.

5 RESULTS

The web-based disease data correlator application has four main view components – the map view, the yearly trend data analysis view, the details view and the correlator view. Each of these views will be discussed in details in the following sub-sections.

5.1 Map View

The main focus of the application is on the Choropleth map displayed in the Map View window. A Choropleth map is a thematic map in which areas are shaded or patterned in proportion to the measurement of the statistical variable being displayed on the map, such as population density or per-capita income. The choropleth map in my Map View shows the value of people having a particular disease encoded as a 6 color sequential colormap where a lighter color represents a lower percentage value and vice versa.

As part of the API package feature, the choropleth map shows the state name and percentage data of disease prevalence in that state's population in an information cloud on mouse hover over a region. The mapping of the data to each state is done using the state name or state code.

Some limitations of this package or view that I felt during the development process are the absence of basic zooming and panning functions and color choice options for background and focused region, which are present in the Google Maps API.

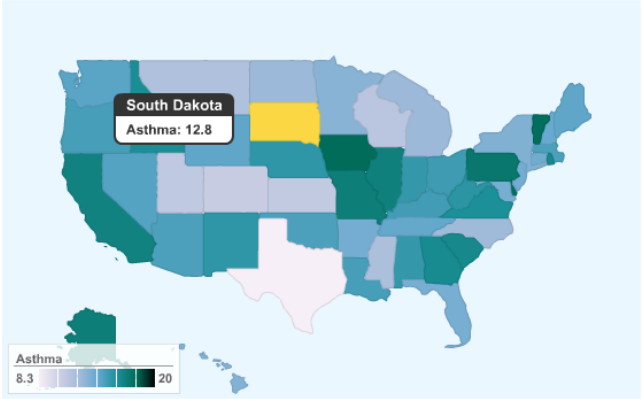


Figure 1: The Choropleth map in the Map View showing the state-wise distribution of the Asthma population of US.

5.2 Yearly Trend Data Analyzer View

CDC provides demographic data from the year 1995 up to the year 2008. Although for most diseases, the data provided for the range of years is not present in reality; these annual data for disease prevalence can be analyzed to find trends of increasing or decreasing infection/prevalence rate for each of the states. I have used only a subset of the annual data provided starting at 2001 and displaying data up to 2008.

The visualization package used for this view is the Annotated Time Series package. It includes a slider at the bottom that can be used to select a data range to display and as a user narrows down/widens the range of years over which he/she wants to view trend the lines plotted for each of the states adjust accordingly in the main time series view. A different hued sequential colormap from the ColorBrewer was used to plot the lines.

A possible extension to this view could be filtering out a subset of the time series lines based on the selection of states in the Choropleth map.

A limitation of this package is that the data are grouped according to the Time co-ordinate value i.e. the data points for a particular year can be selected at a time but a user cannot directly select the values for a particular state which is represented as the data points along a line.

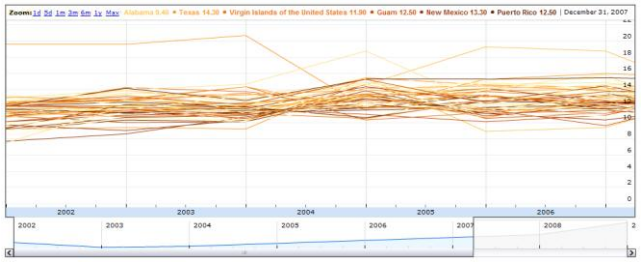


Figure 2 : An annotated time series showing yearly trend data

5.3 Details View

This view is loaded dynamically when a user clicks on a particular region on the Choropleth map. As a result of this event the application fetches the details data i.e. the group-wise demographic data for the selected state from the data source. The demographic data are available for five demographic groups – gender (male and female), race (white, black, Hispanic, other, and multi-racial), education level (up to High School, High School to GED, some post GED level, and College graduate), age (18-24 years, 25-34 years, 35-44 years, 45-54 years, 55-64 years, and 65+ years) and yearly income (Less than 15K, 15K up to 25K, 25K up to 35K, 35K up to 50K, and more than 50K per year).

I have chosen to use the Bar Chart API for the gender based data. The color map chosen is a 2 categorical color palette from ColorBrewer.

The race-based demographic data is shown using the Column Chart API which is essentially a vertical bar chart. Again the color palette has been selected from the ColorBrewer.

The age-wise and income-wise prevalence data are shown using basic line charts where as the education level wise demographic data are shown using the Area Chart API.

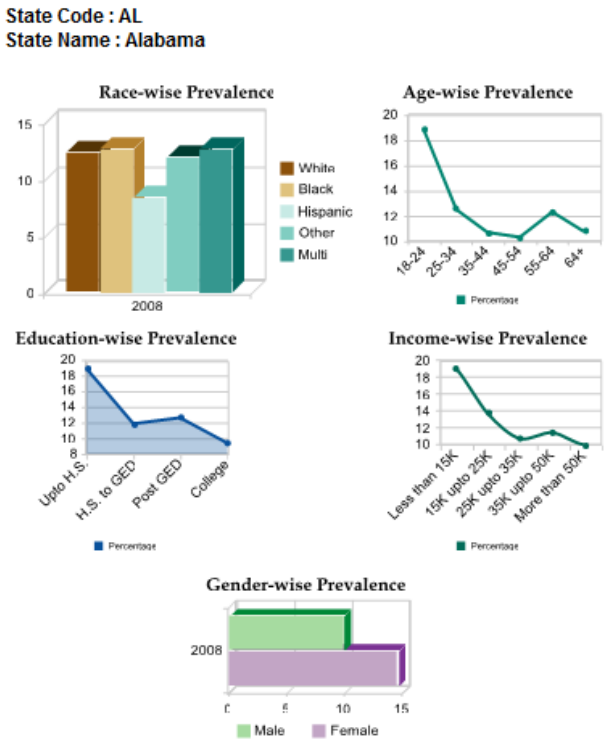


Figure 3: Details view window loaded with the demographic group-based data for the state of Alabama

5.4 Correlator View

This view shows the demographic data trend for all the states in a single chart form. Given this view a user can search for patterns in the prevalence data. For the partial views shown in Figure 4, some hypothesis could be made that Asthma is more prevalent among women than in men and that the Hispanic population has a lower rate of asthma than the white and black population etc. Based on these types of patterns the user can do further data analysis to prove his/her hypothesis.

Initially, during the proposal phase of the project, I mentioned using scatter plots as a suitable representation for pattern detection and analysis. However, during the implementation phase, it became clear to me that due to the categorical nature of the demographic groups, the scatter plots were unsuitable for data analysis. Instead I opted for a line chart to show the demographic data for the states. Another alternative could be using stacked bar charts but I felt that the line charts were easier for finding patterns in the data set.

Another issue relating to the visual representation of the correlator view is the dimensioning of the line charts. To show a clear picture of the state wise prevalence data the charts need to have a longer range along the X axis to plot all the states along the axis. This has made me realize that it would have been better to display the correlator view on a different page or tab altogether.

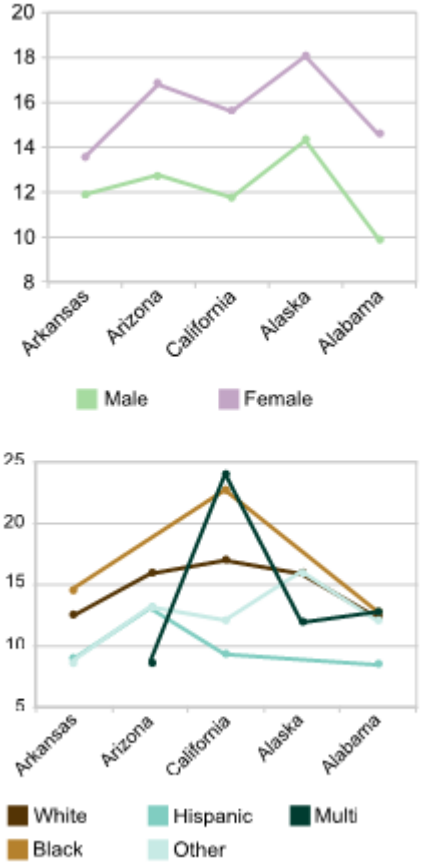
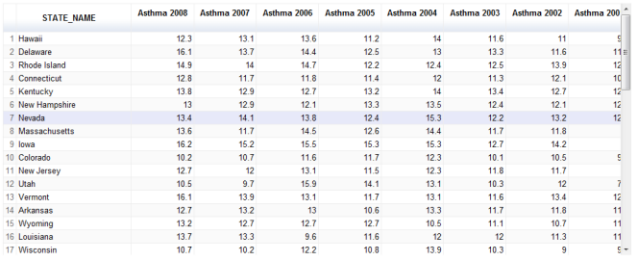


Figure 4: A portion of the correlator view

5.5 Table Browser View

The table browsers are the representation of the data set in the traditional tabular format. I used two table browsers in my application, one in the Map View and one in the Trend Data Analyzer View. These are based on the Table package of Google Visualization API. These tables are sortable by column value.



	STATE_NAME	Asthma 2008	Asthma 2007	Asthma 2006	Asthma 2005	Asthma 2004	Asthma 2003	Asthma 2002	Asthma 200
1	Hawaii	12.3	13.1	13.6	11.2	14	11.6	11	9
2	Delaware	16.1	13.7	14.4	12.5	13	13.3	11.6	11
3	Rhode Island	14.9	14	14.7	12.2	12.4	12.5	13.9	12
4	Connecticut	12.8	11.7	11.8	11.4	12	11.3	12.1	10
5	Kentucky	13.8	12.9	12.7	13.2	14	13.4	12.7	12
6	New Hampshire	13	12.9	12.1	13.3	13.5	12.4	12.1	12
7	Nevada	13.4	14.1	13.8	12.4	15.3	12.2	13.2	12
8	Massachusetts	13.6	11.7	14.5	12.6	14.4	11.7	11.8	
9	Iowa	16.2	15.2	15.5	15.3	15.3	12.7	14.2	
10	Colorado	10.2	10.7	11.6	11.7	12.3	10.1	10.5	9
11	New Jersey	12.7	12	13.1	11.5	12.3	11.8	11.7	
12	Utah	10.5	9.7	15.9	14.1	13.1	10.3	12	7
13	Vermont	16.1	13.9	13.1	11.7	13.1	11.6	13.4	12
14	Arkansas	12.7	13.2	13	10.6	13.3	11.7	11.8	11
15	Wyoming	13.2	12.7	12.7	12.7	10.5	11.1	10.7	11
16	Louisiana	13.7	13.3	9.6	11.6	12	12	11.3	11
17	Wisconsin	10.7	10.2	12.2	10.8	13.9	10.3	9	9

Figure 5: A table browser for the trend data analysis view

6 Discussion

The representations of the disease prevalence rate data discussed so far are effective and widely used. Funded by CDC the developer team of GeoVISTA started the GeoViz project to present an interactive tool to display such data in a user-friendly manner. Unfortunately, I could not find any indication on further progress made on the project. As mentioned in section 4, I started implementing this project using the GeoVISTA studio and explore the GeoTools library for Java in the hope that I could reapply the partial solutions I implemented up to that point in another Java based application. Although in the end I chose to use Google Visualization API working with those toolkits and libraries gave me a glimpse at design of map based software and the tasks involved in map rendering.

In the previous section, I have tried to relate the limitations of each of the views whether they are rooted in the designing of the package or an implementation choice that I have made. Rectification of some of those limitations is discussed in the Future Work section.

7 Future Work

Working on this project has shown me the need for a general and extensive thematic mapping API that could be used instead of depending on any particular toolkit. This issue is particularly important when you consider the plethora of toolkits and applications available for geospatial data representation but scarcity of ones that can be used in conjunction with other free-source tools or APIs. Google already provides such an API that can work with both Google Map and Google Earth APIs. This API provides support for KML based thematic maps of types Choropleth map, prism map, proportional symbol map and pie chart map. I would like to extend on the pie chart map API to show basic bar charts, line chart and later on other more complex charts. Using JFreeChart or Eastwood Chart API with the GeoTools library for Java is another challenge I would like to work on.

Specific to my current project implementation, I would like to handle other events like changing the slider data range of the yearly trend data view, rows or columns selection of tabular data browsers etc. to filter my data and synchronize views accordingly.

Due to the large size of the data set, any change in the dimensioning of each of the charts, maps or views has an effect on the usability of the application. Due to time constraints, an optimal viewing was derived using manual dimension setting both for the Google Visualization API and the web interface. I intend to remove these inflexibilities in a future version to make the layout more dynamic.

A number of other filtering conditions could be added to represent the data in a more readable format. For example a filter that could select one or more states instead of all states at once would improve the degree of comprehension of the trend and correlator views.

8 Conclusion

My project has re-implemented a number of visualization techniques and solutions for a real-world data set that is comprehensible and of relevance to all. My gain from this project has been getting familiarized with different tools used in the field of geospatial data representation as well as generalized visualization solutions as offered by Google. This project has also given me the opportunity to apply some of the infovis design principles that we have learned during the expanse of this course.

References

[1] Chris Brunsdon and Jason Dykes, Geographically weighted visualization – interactive graphics for scale-varying exploratory analysis, IEEE TVCG 13(6):1161-1168 (Proc. InfoVis 2007)

[2] Leland Wilkinson, Anushka Anand, and Robert Grossman, Graph-Theoretic Scagnostics, Proc InfoVis 05

[3] G. Fuchs and H. Schumann, Visualizing Abstract Data on Map, IV ’00,00:139–144, 2004.

[4] M. Gahegan, M. Takatsuka, M. Wheeler and F. Hardisty. GeoVISTA Studio: a geocomputational workbench. *Computers,* Environment and Urban Systems, 26:267–292, 2002.

[5] Diansheng Guo, Mark Gahegan, Alan M. MacEachren, and Biliang Zhou. Multivariate, Analysis and Geovisualization with an Integrated Geographic Knowledge Discovery Approach. Cartography and Geographic Information Science, 32(2):113–132, April 2005.

[6] G. L. Andrienko and N. V. Andrienko. Interactive maps for visual data exploration. International Journal for Geographical Information Science, 13(4):355–374, 1999.

Appendix

Raw Data Format

Column	Variable	[7] Field Length
1	_STATE	[8] 2
16	IDATE	[9] 8
88	_DIABETE2	[10] 1
94	_ASTHMA2	[11] 2