PROCESS BOOK
for

Global Mortality Losses and Causes

(Final Project for CS 171 - DATA VISUALIZATION)

Table of Contents

Original Proposal

Bijish Nedumparambil Lakshmanan
bijishnedumparambill@g.harvard.edu
508-215-8535

Bryan Zadworney
bjzadwor@hotmail.com
318-572-8384

Background and Motivation

We are looking to examine how causes of mortality vary by region, age, and sex. We are trying to determine how people die, particularly looking at where they live and how geography affects how and when they die. In an effort to quantify the effect of the various causes of mortality, we use the years of life lost to measure the affect a disease has on world population. YLL looks at the age a person dies as a way to quantify loss. Unlike other studies which look solely at cause of mortality, we will use YLL as a key factor, allowing us to add a measure of weight to causes which strike earlier in life.

Project Objectives

Provide the primary questions you are trying to answer with your visualization. What would you like to learn and accomplish? List the benefits.

We plan to use the following questions to guide our visualizations, due to the limited amount of time we have to complete this project, we may decide to concentrate our efforts on a few key questions. Our biggest goal is to identify trends in mortality causes, determining the best outlets for scientific study and funding.

1. What are the mortality rates - no. of deaths, years of healthy life lost, years of life lost and years of adjusted life due to the various disability causes? Understanding how people die helps us identify potential opportunities for research.

2. How have the mortality rates changed over time? Ideally we should be able to identify decreases in some mortality rates due to advances in science. This study should also identify those causes which are causing higher levels of YLL as other causes decline.

3. What are major causes for infant mortality? By studying the changes in infant mortality rates we can better understand where we can spend limited funding to achieve the most benefit.

4. How much does death from non-medical causes such as violence, accidents change among age groups and regions?

5. Find which regions/countries have higher violent mortality rates.

6. Examine which causes are most significant by age by sex and by region. Regional differences may affect how countries spend their research money. Some regional trends are to be expected (We don't expect to see much malaria in arctic regions.)

7. Examine which regions are most affected by respiratory diseases? We expect that industrialized countries with less pollution control tend to have high rate of mortality from respiratory related diseases.

8. Which regions have the most number of birth defects? Are the technologically advanced countries showing lower levels birth defects?

9. Which are the regions with the highest cancer rates? Do developed regions have higher rates of cancer?

Data

From where and how are you collecting your data? If appropriate, provide a link to your data sources.

The main data source we plan to use for this visualization is the Institute for Health Metrics and Evaluation (IHME) Global Burden of Disease Study 2010 (DGB 2010).

We need to determine population in the different regions for some of the visualizations. The problem we need to solve is developing filter functions which use the user inputs to determine the data for the visualizations.

Data Processing

Do you expect to do substantial data cleanup? What quantities do you plan to derive from your data? How will data processing be implemented?

The Global Health data is very large, it's broken down by region, sex, mortality cause, and age buckets (but not country). The full data set is too large to download each time the page loads. We will need to generalize the information, decreasing the client computer's workload and shrinking the file transfer sizes. We may need to find population estimates for each region so we can normalize the information to population sizes. We anticipate storing the information on the server, then using ajax calls to get the necessary information based on user inputs.

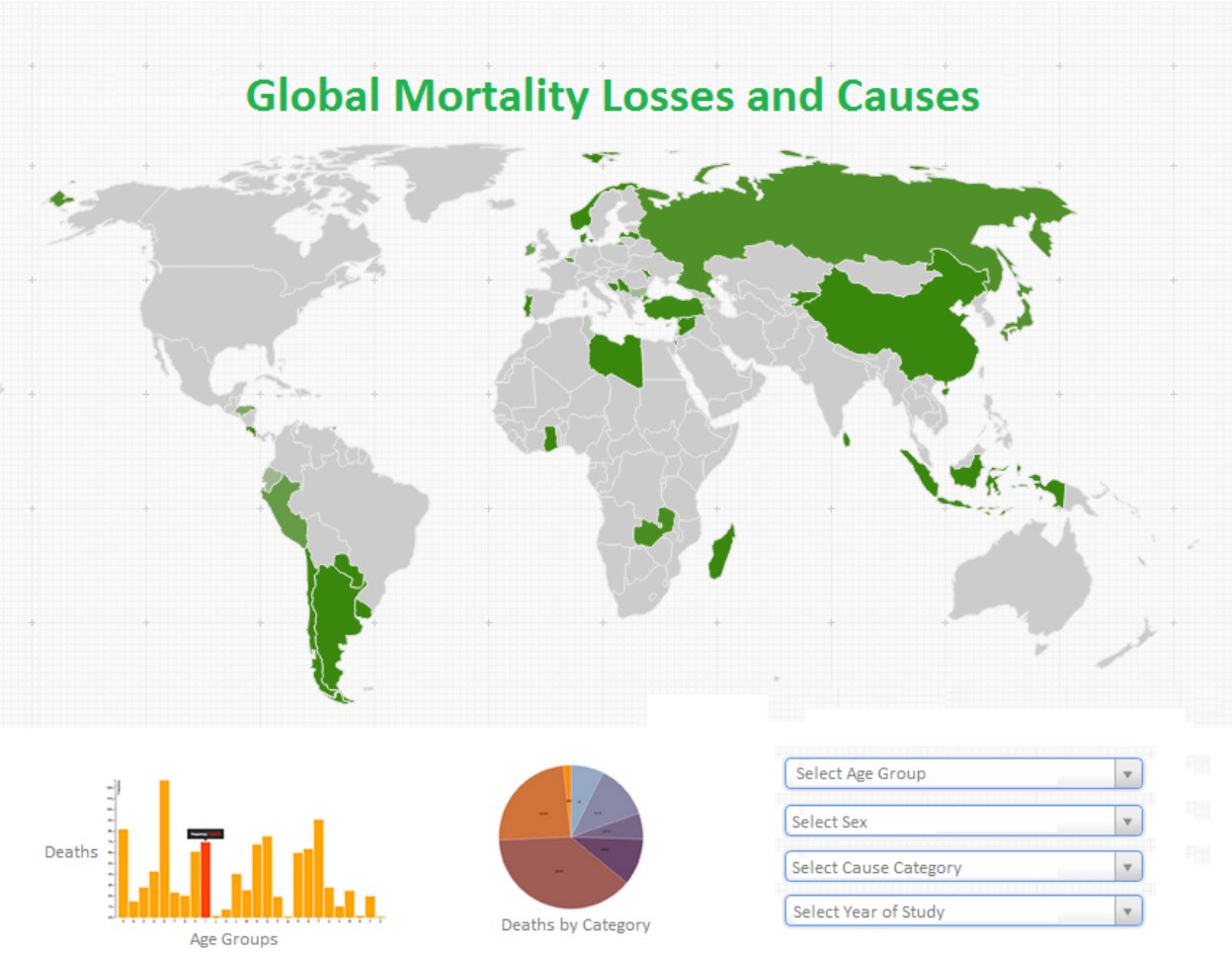
Visualization

How will you display your data? Provide some general ideas that you have for the visualization design. Include sketches of your design.

The site will have a world map, which compares chosen demographics by coloring countries or regions. The page will initially display with some default

demographics, the user will have the option to filter the data. When the user filters the data the page will execute an ajax call, gathering new data and re-drawing the map with the new data. When a user clicks on a region the user views a custom dashboard which gives more information about that region.

We might make the map smaller (top half of page) and show some visualizations along the bottom half of the screen and below the fold.



Must-Have Features

These are features without which you would consider your project to be a failure.

We definitely want to have a world map which colorizes the regions/countries according to selected parameters. We also want to be able to click on a country/region to get more information about that country.

Optional Features

Those features which you consider would be nice to have, but not critical.

The complexity of the individual regional dashboards may change, as the project progresses we will determine how robust we can make the individual dashboards. As we examine the information, we will become more comfortable with how to visualize this information.

Project Schedule

Make sure that you plan your work so that you can avoid a big rush right before the final project deadline, and delegate different modules and responsibilities among your team members. Write this in terms of weekly deadlines.

We plan to use Scrum methodologies to develop this project, instead sticking to a hard schedule we plan to develop a list of tasks. As the project progresses, we will work through the list iterating the project as necessary. We plant to keep the following milestones in mind as we are developing this project:

Week 7 Thursday, March 13: Project proposal due (part of Homework 3)

Week 8 Meet with TA to discuss project, Finalize Data and Plan

Week 9 Begin drawing world map with GeoJSON (HW4 Due)

Week 10 Begin displaying Individual Dashboards

Week 11 Thursday, April 10: Functional project prototype due

Week 12 Project review with the TFs

Week 13 Finalize Project Create Screencast

Thursday, May 1: Projects due (including screencast)

Final project proposal feedback:

From Bryan's HW 3:
Comments P1:

"Nice proposal. The idea of modular vis is interesting, but not linking, etc"

From Bijish HW 3:
Comments P1:

"Good job, implementing your sketch would be a good project scope to start with"

From Project Feedback:
Hi Bryan, Bijish,
I'm Julie Zhang and going to be your final project TF. You two had a good proposal, however I believe that given the previous work we've done in the course, and that the final project should be at least double the scope of a normal homework, I think the minimum "must-have" features of your project should be your project sketch: the global map and the chart dashboards.

For FP1:

** The data collection, cleaning and wrangling has to be done. The data must be in a form that can directly be used by the visualization. It's OK if the dataset isn't complete (e.g., there will be more rows in a file), but the structure of the file and a reasonable sample must be there.*
** There must be a visible and working visualization of their major views. Maybe not all, maybe not in quite the complexity and not with all the interaction. But it should be there and it should be in D3.*
** Also submit your in-progress process book.*

In particular, for FP1 your visualization should have the global map portion completed, encoded with data, and interactive (clickable, zoomable at least). At least one dashboard chart should be done.

Please also think of some additional optional features beyond your current plan. Perhaps filters?

Good luck, and let me know if you have any questions.

Julie Zhang

Regional GeoJSON:

The first step of our implementation was to visualize the 21 geographical regions represented in the study. We originally expected to find geoJSON or some other depiction of the regions which we could use to draw a choropleth. We quickly learned the 21 geographical regions used for this study do not correspond to any of the "standard" regions used in other studies, which meant the regional geoJSON did not exist, we needed to generate them ourselves.
We began the process by downloading international boundaries from:
http://www.naturalearthdata.com/downloads/50m-cultural-vectors/

We then used the information from the following link to determine which countries were in each region:
http://ghdx.healthmetricsandevaluation.org/country_profiles

We used the vectors representing all the countries, and a list of the countries in each region, to build a geoJSON file for each region. We used software called QGIS, which allowed me to "select" the countries in a region, from the cultural vectors and save their border information to a geoJSON file representing all of the countries in the region. We then edited the geoJSON files adding an attribute identifying the region to the geoJSON. We then needed to find a way to draw the 21 regions into one map, so wel combined the 21 separate regions into one geoJSON file, which we used to draw a map of the Globe.

We initially drew the world map, and realized there were some problems with our geoJSON. A few of the African countries represented in the survey experienced a civil war, and split into separate countries, our geoJSON represented the latest geographical regions, so there were holes representing the new countries. We decided to represent these countries as one entity in our map, since they were one entity at the time the data was recorded. Once the continent of Africa was complete, we concentrated on the northern hemisphere where something was off. After studying some maps, we realized the island/country of Greenland was not represented in the survey, and is not showing in our map. We've reached out to the originators of the survey to find out which region Greenland falls into, at this point Greenland is not represented in our visualization.

We now had a world map, however we still represented countries, not regions. We needed a way to color the regions, and highlight them on mouseover. We chose to add a attribute with a class value of the region code to each country path. When the user mouses over a country, we use d3 to get the region code out of the data associated with that country, then color all country "paths" with that class yellow. When the user moves the mouse off the country/region, the color reverts to the original color. This allowed us to show the regions in a clear/concise, and interactive method.

Data Size

The original the Institute for Health Metrics and Evaluation (IMHE) data set is many gigabytes large, IMHE is still working on a web service to provide access to the data. Instead of offering a robust web service,IMHE provides a number of .csv files available for download. We wanted to explore global health trends at a global level, so we decided to work with their regional health data set.

The data set we chose to work with showed data for 21 regions, with over 240 diseases, multiple data points per disease, broken up by age, and sex. The data set represented 3 different years of study. The original .csv file was approximately 220MB with over 832,000 rows, which was still too large to work with using d3.csv to process and import the data.

We decided we needed to slim down the data. We initially decided to reduce the number of columns in the data set, reducing redundant and unnecessary columns, but retained all of the rows. This slimmed the data slightly, but we needed to do more. We examined the data, when we drew our choropleth we realized that the regions containing India and China almost always showed the largest values due to their huge populations, so we decided to get rid of the columns which dealt with absolute numbers, instead choosing to work with columns dealing with population normalized death and injury rates (rate per 100,000 people) removing these columns slimmed the data set down to close to 75MB, but it was still too large to upload to github, and it took way too long to load the page. Our next step was modifying the data, we removed all the region names, replacing them with a 3 letter code (South East Asia became SEA). This change got the dataset down to approximately 60MB (still too large for github, and we needed to get the data into github so we can submit FP-1). We then removed the commas out of the strings (1,345,345 became 1345345) This got us down into the 50MB range, once we changed the year values from 4 digits to 2 digit s(1995 became 5, 2010 became 10) this got us below the magical 50mb threshold we needed to submit our data to github.

Our dataset is still too large, if you load the page using a local server the data loads within a few seconds, however loading the page over the internet takes over 45 seconds (1:30 at a Starbucks). We've decided there are a few more things we can do to load the page quicker:

1. We can slim the data set by converting the diseases to a 3 letter code, like we did with the regions.
2. We've broken the data set into 22 smaller data sets. When the page loads the "global" data set will load. The first time the user selects a sub-region, the data set for that region will download, and get appended to the dataset. Loading the page in this way allows us to rapidly display the page, while still giving us the flexibility to display the full data set as required to meet the users' needs.

Data Manipulation Tools

We've used a number of different programmatic tools to split the data set, we've documented some of the methods below. This example shows how to split the large data set into Male, Female, and Both data sets based on the "sex" column. We

eventually decided to break the data up by region, instead of sex, however the technique is the same, we simply use region name instead of ",Male," and male.csv.

Splitting data files based on sex to reduce file size-

Use the git bash shell to run the following commands to split the data files based on sex

```
$ grep ",Male," full.csv > male.csv
$ grep ",Female," full.csv > female.csv
$ grep ",Both," full.csv > both.csv
```

Use word count (wc) with -l option to do a line count to make sure the rows count tally up

```
$ wc -l *.csv
 283826 both.csv
  280367 female.csv
  832906 full.csv
  268712 male.csv
1665811 total
```

prefix the header row `"cause_medium,region_name,year,age_name,sex_name,death_abs,death_rate,YLL_abs,YLL_rate,YLD_abs,YLD_rate,DALY_abs,DALY_rate"` to the newly split files.

Filtering Data

D3 offers a filter function, unfortunately this function is called each time a visualization is generated, which would mean the filter would walk through the dataset multiple times in order to draw the visualizations for our page. The dataset used for this page has the potential to be extremely large, we decided to write our own filter function, looping through the large data set producing smaller data sets which the visualization functions use as necessary. Anytime the user changes the form, the filter function is called. This filter function reads all the form values into a JS object, then loops through the dataset building smaller datasets. As we develop new visualizations we simply modify the filter function to develop a new dataset, or we use one of the existing datasets with the new visualization.

Using the filter function to loop through the datasets speeds page transitions, and simplifies the code.

Meeting Notes

The following notes document the brainstorming points we took away from some of our in person meetings These notes do not document all of our meetings, as some meetings ended with programming tasks, which were documented in the code and subsequently removed. These notes are a stream of consciousness, we typed and organized them as we mused over the project, they exist now as they existed after the meetings. We also met daily over google hangouts, those meetings were usually smaller, and not documented here.

4/2/2014:

Brainstorming points for later elaboration

- What data will be plotted on the map?
- How can the death causes be listed on the page without taking up a lot of real-estate?
- Will there be comparison of regional data?
-How can multiple selections of causes/age-groups/sex/year be accommodated? (usage of checkboxes)
- How can the data from various years - 1990/2005/2010 be used? (showing changes over the years)
- What is the default view?
- If there is shading, what is the basis for the color?
- Data Structures?
- How will data be loaded from the csv file? In what order?
- What are the various filters?

4/8/2014:

To explore-

- What other visualizations to include?
- What color schemes to use?
- Add different visualization layouts for the user to configure as needed

Data structures-

- Different datasets to hold region-names/cause-names/age-groups/metric-names
- How to hold data optimally to be available for all of the visualizations?
- Loading datasets for different regions only on need basis and not to load all the data initially. As the user selects different regions, it will be loaded into d3. Once the data is loaded for a region, it is not removed so it loads only once.

Data Cleansing-

- Using short names for region names
- Splitting up of the full data CSV file into separate CSVs region-wise

- Making the following changes to the original CSV files to reduce file size and optimize data load times

1. Removing unwanted columns from the original CSV
2. Shortening the region names TLA (3 letter acronyms) region codes,
3. Replacing 4 digit year with 2 digits,
4. Using short cause name instead of the medium cause name
5. Shortening sex names (from Male/Female/Both to M/F/B)
6. Removing commas (000 separators) in numbers

4/10/2014

Considering building a line graph showing how the chosen data changes over the 3 time ranges (One of our original objectives).

Considering giving a dropdown box so users can choose what shows in the large top display, this will allow the users to show something “big”, and use the form to change the data.

The rest of this meeting was spent doing the Design Studio 3 discussions, as described below.

Design Studio 3 - Feedback

We met with Lana Nelson (shvetusya@gmail.com) and Charles Bandes (charley@charlesbandes.com) on 4-10-2014

Feedback:

- Overall excellent project, we're on the right path to display the data using very clear methodologies.
- When we discussed the size of the data we talked about setting up some node.js or server side alternatives to serve the data on demand, and agreed it was outside the scope of the project. We also talked about how we were going to break the data into regions, then load the regions on demand. There was some discussion about automatically loading the regions in the background, plus downloading the data on demand if it has not downloaded once the user requests a region.
- The projection we chose to use, is the correct projection to use for a Choropleth, however it's still weird, we might want to consider allowing the user to choose from different projections.
- The bar charts are still a little difficult to understand what they are as it was work-in-progress and we all agreed we need to title them.
- They liked how we used a modal window to zoom in on the smaller images.
- They liked the idea of being able to choose the image in the top where the map is.
- They think we should explore zooming the map into the chosen region, or centering the map on the region.
- They recommended exploring how we could use sex as a second dimension in the bar charts. They recommended using a stacked bar chart, or side by side bars to show the two different sexes. They also thought we could show the 3 different years in one chart if we kept the number of x axis values small enough.
- They suggested using one color, and controlling the saturation instead of using a two color range.