

Project Proposal

Sean Smith, Viviana Yee
{swsmith, vivyee}@bu.edu

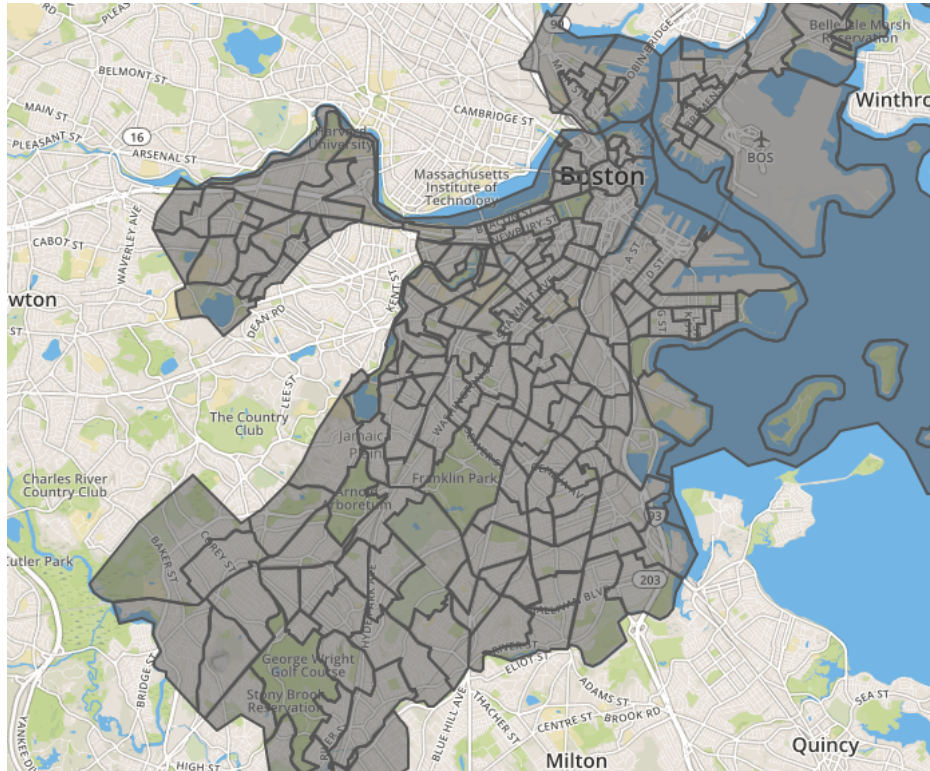
15 March 2017

1 Problem Definition

The city of Boston maintains an open data initiative that provides datasets on lots of different city services <https://data.cityofboston.gov/>. We both have experience taking large amounts of data and making conclusions based on trends and correlations between different datasets. Based on Viviana's experience in Andrei Lapet's Data Mechanics Course and Sean's experience with New York City Taxi data and geographical zone information we've decided to look at Boston obesity data and locations of nutrition centers and the distance between them. Specifically we want to model MBTA transportation options from different neighborhoods (for which we're able to calculate the obesity rate) to nutrition centers. We aim to model the distance in order to answer our hypothesis "Do neighborhoods further from nutrition centers suffer from greater obesity rates?".

2 Data Sources

First we needed to define the geographic areas that we'll be modeling, to do so we've downloaded the 2010 census tracts dataset. This breaks Boston up into unique tracts that each have approximately the same number of residents. Sean worked on writing a point in polygon algorithm which decides if a coordinate pair is contained in a neighborhood last semester for his Multi-party Computation Class. This algorithm can be modeled using WFF's.



Next, we took Obesity data from the CDC, Center for Disease Control, and Nutrition Program data from the city of Boston. We filtered these two data sets for longitude and latitude data, so it is valuable for our problem.

Finally, we called the MBTA API endpoints for MBTA stop data. We did this by getting routes for the bus and T. From that, we individually got all stops for each route because there is no API call for all stops in the MBTA system.

By modeling our data with propositional logic we hope to answer the following questions.

3 Methodology

1. Find shortest path to all nutrition programs and find shortest out of that
2. Pick a neighborhood with a nutrition program that is geographically closest, and find shortest path to that

3. Combine (find all neighborhoods within certain distance and calculate shortest path for programs within the radius)
4. Find shortest path from every neighborhood to every nutrition center (depends on running time)