Application Project Report
Jonathan Gluck (jdg@umd.edu) and Gregory Kramida (gkramida@cs.umd.edu),
CMSC 734, CS Dept. UMD, Graduate Students

Data

We had two primary datasets for this project. One was the inflation-adjusted income from the IRS tax return data, for years between 2000 and 2010 (inclusive) at every zip code, another was the Dogs of NYC dataset, containing birth year, zip code, breed, and several other attributes for every dog on file. These two data sets can be joined on year and zip, thus we only looked at NYC zip codes for the income set. We also obtained various peripheral data from the NYC OpenData project, such as locations and names of cafes, electronics stores, and laundromats for every zip code. Unfortunately, this data had no time variable, so it could only be joined locationally by zip code to the other datasets. Most other data from the NYC OpenData project was even less applicable. For instance, math scores were collected for year and grade rather than location, while Farmer’s Markets’ locations could only be looked up by street intersection where they were located (no zip code or even street address). We were also going to look at US census data, but it was not available due to the partial government shutdown.

We ended up getting the data for the zip codes we needed from melissadata.com, literally scraping the data zip-by-zip from html tables and putting into one large csv file with a script we wrote in python. Unfortunately, the site imposes a limit of 25 lookups per day per IP address for unregistered users. The limit is higher for registered users, but it wasn’t time-efficient to integrate emulation of user credentials into our script. Instead we collected this data over several days and using multiple IPs. Our final “income” dataset included a total of 2292 records and contained data spanning 11 years for 209 zip code areas. Data for some of the 225 zip codes we originally targeted was either entirely or partially missing from the melissadata website.

The dog registration (“dog”) dataset we retrieved directly by downloading the csv file. It contains 81542 records total, which also span those 225 zip codes we targeted for the income. The dog birth dates vary from 1966 to 2013, but we found that the bulk of the data (83%) also spanned 1999 through 2010 (note the year offset - this is due to the tax returns actually accounting for the income in the previous year they were filed). We had to make the assumption that every dog was most likely registered in a particular zip within the same year it was born to temporally tie dog ownership to zip codes.

Application Critique

All of the datasets were imported into a PostgreSQL relational database, where they were cleaned, formatted, and indexed (the above-mentioned figures represent totals after cleaning). We decided to give the popular Tableau software a go in our analysis. After we installed the
missing drivers, it had no problem with connecting to PostgreSQL. One of the first HCI-related inconveniences we’ve noticed was that Tableau offers no way to retain connection information between workbooks, so if an earlier connection needs be repeated for a new file, the user has to type in everything from scratch.

The Tableau software offers support for a wide variety of data management systems. It does not store any of this data locally. It uses a driver to build the queries required by the user interactions. This lead to slow action time when updating views; as the new view had to be queried from the server.

We found that Tableau was somewhat frustrating to use. We would sink significant periods of time into trying to discover how to accomplish a simple task, only to ultimately decide to modify our data in PostgreSQL. This may be due to our mindset; as we are familiar with PostgreSQL but not with Tableau.

Tableau offers “calculated fields” as a very powerful tool. For instance, when we were examining the income of a particular zip-code over time, we were able to easily write a calculated field that had us examining percent difference in income instead. A major shortcoming of this powerful feature is that users cannot combine aggregate and non-aggregate operations. For instance, if one wanted to calculate a field which represents deviation from the overall average at a given point, Tableau does not support this.

Overall, Tableau is a powerful tool; however, even its experts seem to have trouble wrangling all of its requirements. It offers extensive support for existing data management systems, which is why we chose it; however, due to the way in which this support is implemented, it lead to a sub-par user experience.
Findings

*Income Patterns over Time*

One of the more significant discoveries that we made was that the zip codes with higher overall (inflation adjusted) incomes followed a similar pattern over time amongst themselves, but this pattern was radically different from the overall pattern and the average. The chart on the left shows the year-to-year differences in filed income for each particular zip, where the degree of deviation from the norm is marked by how blue the line is. The average pattern is marked in green, and you can tell that although it’s different, it is still offset significantly by the rich.

The coordinated view also contains the geographic zip representation (the only piece of census data we were able to get, due to it being cached by a third-party server). The color represents the average inflation-adjusted income (greener is higher). We also figured out a way to add “page controls” - a slider that allows us to traverse the years and see how the map changes from one year to the next.
Highlighting individual zips and viewing their pattern over time also enabled us to find several “anomaly” areas, which we then researched individually. For instance, the bay area highlighted above at first adheres more to the “rich people” pattern, but around the time of the recession the zip starts conforming more to the majority. The map allowed us to determine that zip with the extraordinary spike in the income is actually limited to just a single city block. What’s more interesting, it turns out there are no actual people living there - it’s a zip code reserved for P.O. boxes which are apparently used mostly by millionaires.
The visualization above has income categories along the abscissa and dog density (number of dogs per return) along the ordinate. Each column is split into dog breeds by color. Two things immediately grabbed our attention: exactly at the millionaire category, there is suddenly an overwhelmingly large amount of poodles, while at the 20k level people are getting lots of lab retriever cross-breeds.
Another interesting find was comparison of the relationships between the density of different types of amenities, namely electronics stores, cafes, and laundromats, to average income in each area of the city. It turns out that cafes are the most strongly correlated, and, as the visual trend-line suggests, grow in quantities together with the amount of money in the pockets of nearby residents. In the meantime, electronics stores follow a similar trend but to a much lesser degree (and lesser significance). Finally, the laundromats mostly camp out in the poorer areas (where cafes are scarce), but there are a few outliers downtown - possibly, the dry cleaners where businessmen leave their coffee-stained neck ties.

Data Source References
NYC OpenData Project: [link](link)
Melissadata: [link](link)
Dogs of NYC dataset: [link](link)