Nicholas Choo
nchoo

CS229 Project Milestone

Finance & Commerce: Where should I live?

Background

Data USA is a collaboration between Delloite, Datawheel and the MIT Macro connections group designed to structure US public data into an easily accessible format. It allows systematic access of many US government databases, and allows the aggregation of data by geographical location.

With declining job loyalty and decreased tenure, changes to the traditional job market mark an increased independence by employees and increased likelihood by millennial to leave their job within five years [1]. The interest in changing jobs is a conundrum without available data, and assumptions that there are better opportunities elsewhere causes an individual to consider changing jobs may not necessarily be true.

This project aims to use a data driven approach to drive individuals to understand their markets in order to maximize opportunity. By combining data about trends in demographics and work from Bureau of Labor and Statistics, Bureau of Economic Analysis, American Community Survey and the County Business Patterns, the data aims to extract trends from the large number of features related to each geographic area

Related Work

The area of geodemographics is a heavily studied. Bacao et al. [2] and Blake et al. [5] have presented a comparison of self organizing maps (SOM) versus K-means clustering, claiming that SOM is less prone to local optima in census data and are able to remove the assumption that geographical divisions have nothing in common with one another as a property of the SOM update function. Jacobson et al [4] have derived a fuzzy weighted clustering algorithm with a special distance metric that take in account the spatial location of a region in relation to others within a cluster. All of these algorithms however consider data from Portugal and Britain, and not within the United States. Expectation Maximization algorithm is also not used with any of these works.

Goals

This project aims to provide individuals a recommendation of areas where someone of their skillset and interests would thrive. We hope to use clustering to determine the best match for an individual. Predictions would aim to match skillsets with job availability, average ages and cost of living with expected values for the subject. In addition, with the large availability of demographic information, we could make less obvious predictions such as race and wage relationships for specific occupations, percentage income above average for occupations, weighing these against standards of living or provide general recommendations in each area for commute or health care.

The input to the algorithm is a person vector. We perform clustering of the data from dataUSA API, and predict the most suitable cluster for the target individual. To get a top list of cities, we ask the individual to order their priority of the following secondary parameters,

We then create a weighted objective function that will be used to give each city in the target cluster a specific rank. The user is then returned with a list of top cities that matches their profile and recommendation.

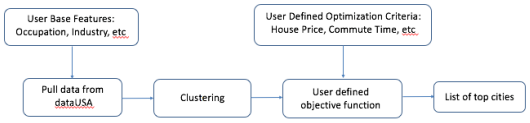

Figure 1. Model of the data pipeline used to generate results for this project

Features and Dataset

Data that features were pulled came from 4 different data sources. They are the American Community Survey, Bureau of Labor Statistics,

County Business Patterns and County Health rankings. We found that the secondary clusters not as well suited for clustering, thus the introduction of an objective function to rank the cities.

Many features were informational, or not relevant to fitting the user to the data. An example is average wage. Since the user is unable to dictate wage, this parameter is useful to be maximized during the fit, but user input is not relevant for this use case.

The dataset contained hundreds of metrics for thousands of geographical regions for thousands of occupations and industries. To simplify analysis in the report, our model was applied only to Electrical and Electronics engineers working on Computer Systems Modeling.

For the data we used PUMAs – public use microdata areas with population > 100,000. This gave a dataset with m = 2079. To further clean the data, we ignored all categorical and non numerical features. We also removed features such as margin of error on average populations and wages, as this data was not core to the clustering process.

We found part time job features to also be incomplete, and as a result we removed these from the dataset. After fitting the data, it was realized that data for areas with under 100 individuals was often outliers, and for the intended goal of the project, not a good result, as it may be difficult to find a job in an area with very few employees.

Since the magnitude of the features varied largely, each input was normalized by dividing each value by the average of the feature. Based on plotting the features, we found that most were normally distributed, and thus dividing by average provided a means to level contributions from each feature when calculating reconstruction loss.

Methods

In the algorithm, we implemented two forms of clustering, Expectation Minimization (EM) algorithm and K-means clustering. Principle Components Analysis (PCA) was used to visualize data to help with the discussion portion, however by itself it did not fit in well with the framework of the model so it was not used for predictions.

For both of the algorithms listed, we used reconstruction loss as the preferred metric to quantify algorithmic performance. This is calculated by the distortion function

$$J(c,\mu) = \sum_{i=1}^{m} ||x^{(i)} - \mu_{c^{(i)}}||^2$$

Equation 1. Reconstruction loss, given by the squared distance from a point to the nearest cluster [3]

For K-means, it was necessary to determine the optimal number of clusters in the data. A plot was generated from k = 1 to 30, with 10-fold cross validation ran. The results are plotted in Figure 2.

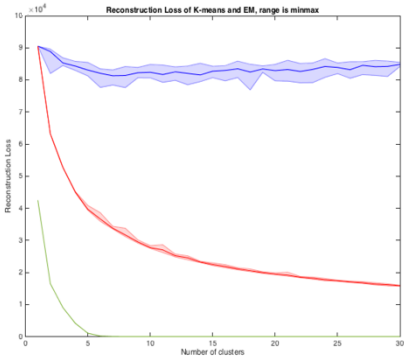

Figure 2. Reconstruction loss statistics versus number of components for K-means (red), EM algorithm (blue) and PCA (green) for the given data

Similarly, for EM algorithm, the number of gaussians was also varied from 1 to 30. For ease of notation we will also label this number k. The results are also plotted in figure 2.Finally, to

study the different types of algorithms, a reconstruction loss for PCA was also included, however in this case, k was equal to the number of components used to reconstruct the data.

Reconstruction loss for PCA was calculated differently; the data was transformed using k number of principle components, and then transformed back using all of the principle components. The difference between the original point and the transformed point was then squared and added for all the data.

In the data, reconstruction loss was a minimum for EM at k = 6. Reconstruction loss decreased as k increased for k-means, however concerns for overfitting lead to searching for the point where reconstruction loss did not significantly improve. This point was determined to be about k = 9, though it was not unambiguously chosen.

From this data, bootstrapping was then used to quantify bias and variance of each algorithm. 1000 models were trained, and tested, with a ratio 75 to 25 for training set to validation set. These parameters were selected randomly. Plotting the distributions resulted in the following histograms:

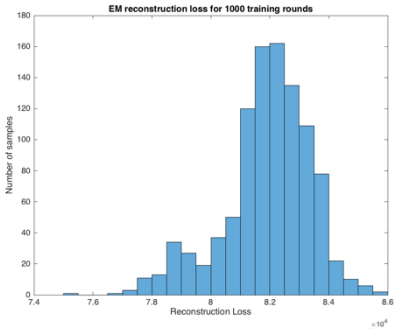

Figure 3. EM reconstruction loss for 1000 training rounds with k = 6, Mean = 8.25, StDev = 0.23

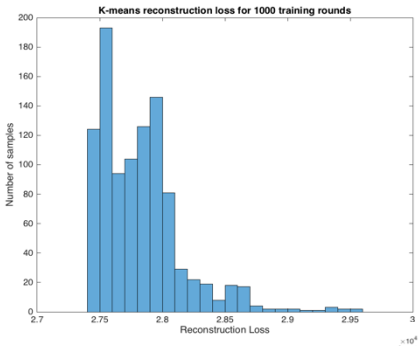

Figure 4. EM reconstruction loss for 1000 training rounds with k = 9, Mean = 2.78, Stdev = 0.04

Another visualization of the data was using PCA dimensionality reduction to analyze the output of the algorithms. Choosing the first two principle components, transforming the data then labeling the data produced the following graphs

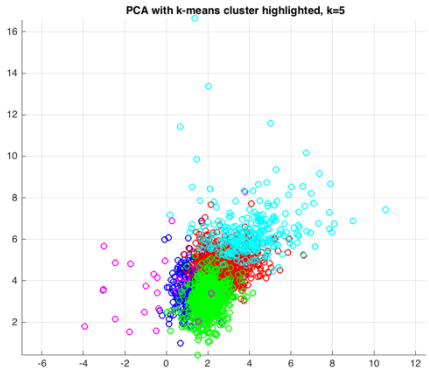

Figure 5. PCA visualization of k-means clustering with k = 9

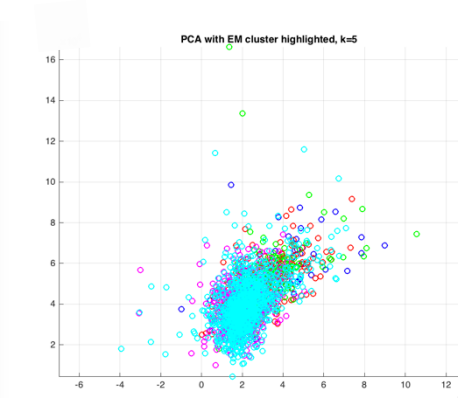

Figure 6. PCA visualization of EM algorithm with k = 6

Finally an objective function was created by enumerating the preference of the secondary features and placing in a vector s. This would be multiplied by feature vector x' that contained the corresponding secondary features for a given city. A rank of each city could thus be determined by the following equation

$$R = s * x'^{(i)}$$

Equation 2. Score equation used to calculate the score of a city in the cluster

An arbitrary vector was chosen, prioritizing house price, income, commute time, then work hours.

	K-means	EM
1	Austin, TX	King County (North East), WA
2	Greater Bellevue City, WA	Huntington Town, NY
3	San Diego, CA	Walnut Creek, CA
4	Alameda County, CA	Santa Clara County, CA
5	Huntington Town, NY	LA Calabasas, Malibu & Westlake, CA

Table 1. Top 5 cities predicted from each algorithm.

Discussion

Clustering as applied to the base data of the problem was formidable and seemed to suit the application for the project. The resulting predicted cities matched articles predicting best cities to work for Electrical Engineers or in the industry Computer Systems, however no articles contained both criteria.

Performance of k-means in this application was generally better than that of EM algorithm. First, considering Figure 2, we see that k-means and EM algorithm have very different behaviors as the number of clusters increases. EM algorithm seems to have a valley, where reconstruction loss is a minimum, whereas k-means has an 'elbow' shape, allowing the model to reduce reconstruction loss further as the number of clusters increase. Based from this information, EM seemingly gets trapped in a local minimum, preventing better clustering.

Looking at the PCA visualization, we can visualize the outputs of the given algorithms from Figures 5 and 6. Data belonging to a cluster in k-means is visually tighter, and closer together. Individual clusters are recognizable from the data. EM clusters however seem to stack on top of each other. This behavior indicates that perhaps the gaussians are not selecting the proper features to predict from, with clusters in a different dimension than the features that present the most information.

The histograms in figures 3 and 4 provide additional metrics which also confirms the better performance of k-means compared to EM algorithm. When comparing the reconstruction loss, we determine that lower mean suggests lower bias, and lower standard deviation suggests lower variance. Thus, the performance of k-means is better than that of EM algorithm.

Aside from the numerical comparisons for the performance of data, determining the performance of the algorithms based on qualitative means was much less deterministic. An interesting observation was that Huntington Town NY was not found in top areas to live for Electronic engineers. As a relatively smaller area with 628 working electronic engineers, but with an average wage of 170,000 and an average age of 45, the presence on the top list suggests clustering provides a more systematic method of filtering through census data.

However, clustering was not without disadvantages. The optimal number of clusters for cities based on our data was smaller than expected, and it tended to map person vectors with where they were expected to live instead of where they wanted to live. It was difficult to map user preferences to what we labeled as secondary features, since an individual usually does not have meaningful input regarding ideal number of doctors in their desired area, percentages of low birth rates or other features.

We attempted to overcome this issue by designing an objective function to rank cities based on user tolerance to certain data. This also had the advantage of isolating the top cities based on user preference, selecting out the most desirable cities within a cluster that the algorithm predicted a person to live.

Conclusion and Future Work

In our application, K-means is better suited than EM algorithm for clustering census data. It was less likely to overfit, less likely to get stuck in local minima and also produced better results. Based on results from k-means clustering, there about 9 different archetypes of cities, based on age, wage and number of electronic engineers. This suggests that the variation in city types is generally not very high.

Additional work that could be done for this project would be to predict cities to live in other countries based on the models built from this work. This would require the same granularity and similar divisions of features as provided by dataUSA from other census data.

More datasets could also be added, for example immigration data or hiring data could be used to also include information about the likelihood of finding a job in a certain area.

Additional investigations into why EM performs significantly worse than k-means algorithm in this case would be priority and also trying other unsupervised learning models or combinations, such as a self organizing map or PCA + clustering

References

[1] “The 2016 Deloitte Millennial Survey.” Delloite Touche Tohmatsu Limited. 2016 [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/global/Documents/About-Deloitte/gx-millenial-survey-2016-exec-summary.pdf

[2] B.Fernando, V. Lobo, and M. Painho. "Self-organizing maps as substitutes for k-means clustering." *International Conference on Computational Science*. Springer Berlin Heidelberg, 2005.

[3] A. Ng, CS229 Class Notes 7a [Online]. Available: http://cs229.stanford.edu/notes/cs229-notes7a.ps

[4] Z. Feng and R. Flowerdew, “Fuzzy geodemographics: a contribution from fuzzy clustering methods”, in Innovations in GIS 5 , S. Carver, Editor. 1998, Taylor & Francis: London. p. 119-127.

[5] S. Openshaw, M. Blake, and C. Wymer, “Using neurocomputing methods to classify Britain's residential areas”, in Innovations in GIS , P. Fisher, Editor. 1995, Taylor and Francis. p. 97-111.