Understanding Wealth in New York City From the Activity of Local Businesses

Vincent S. Chen
Department of Computer Science
Stanford University
vschen@stanford.edu

Dan X. Yu
Department of Computer Science
Stanford University
dxyu@stanford.edu

Abstract

In this work, we aim to understand the relationships between local businesses and the demographics of communities that surround them within New York City. Understanding demographics in relation to business activity is important for applications both in business and social sciences. In our approach, we aim to predict demographics using social-media check-in data. We construct a novel dataset that combines FourSqaure check-in data [1] with census data from the 2015 American Community Survey [2]. Afterwards, we manually construct features based on the various check-in categories and timestamps. We test a number of models on these features and we achieve 41.03% accuracy on an income classification task using Gradient Boosting Trees, compared to 25% accuracy from random guessing.

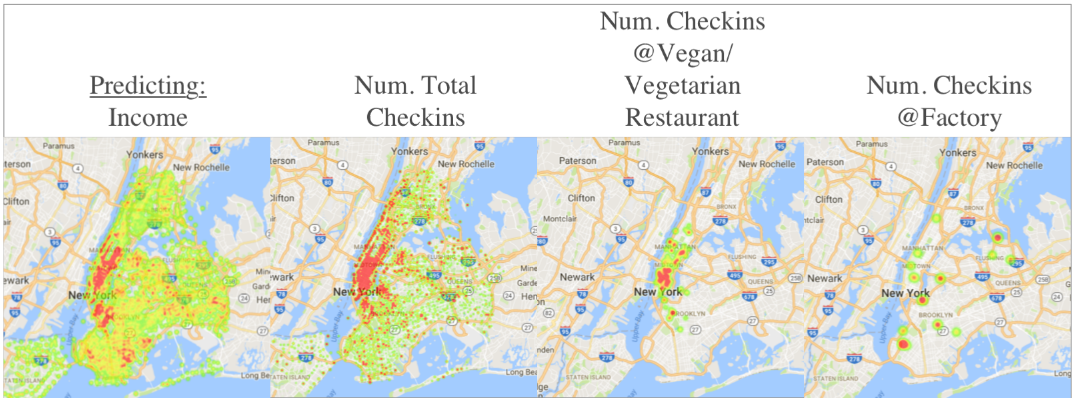
1 Introduction

In the business world, an understanding of the local demographics of a city allows enterprises to better cater their products and services towards those individuals who live there. Researchers within the realm of social sciences can also seek to leverage an understanding of businesses to discern qualities about the people living near them. For example, research in this space can reveal how different categories of businesses can affect gentrification.

Additionally, given how quickly large, metropolitan cities change, it becomes inefficient to rely on costly and infrequent government census services to provide demographic information. To address this problem, we have created a dataset and manually engineered features for the prediction of demographics in New York City. We show an approach for using this dataset to model wealth, based on income, for individual census tracts in the city.

Our approach is informed by prior assumptions about the dynamics between businesses and the people who live in cities. In dense urban centers, we expect people and businesses to interact relatively frequently, due to their close proximity. As a result, we believe that locations of businesses within city are potentially reflective of the people living in those local areas.

We are particularly interested in New York City (NYC) because it has a high density in both businesses and people. To find relationships between regions of a city, we map check-in data from Foursquare [1] to New York City census records [2]. We then manually engineer features based on combinations of business categories (e.g. Arts & Crafts Store, Medical Center) and time (e.g. Mondays, '8am to 12pm'). Finally, we model the relationships between these features and wealth in a classification problem, for which we predict four separate quartiles of wealth (e.g. top 25%, top 25%-50%), and report an accuracy of 41.03% accuracy for this task.



Predicting: Income

Num. Total Checkins

Num. Checkins @Vegan/ Vegetarian Restaurant

Num. Checkins @Factory

Figure 1: Heatmaps showing the regional distribution of our prediction target (income) and relevant features. Red areas indicate higher income for its respective graph, and higher density for check-ins.

2 Methodology

2.1 Datasets

2.1.1 NYC FourSquare Check-in Dataset

We use a FourSquare check-in dataset from Yang et. al [1]. It consists of 227,428 New York City check-ins, collected between 12 April 2012 and 16 February 2013. Each check-in contains a timestamp, venue category (e.g. Arts & Crafts Store, Medical Center), latitude/longitude coordinate, `userId`, and `venueId`.

2.1.2 NYC Census Data (2015)

Our census dataset comes from the American Community Survey's 5-10 year estimates from 2015 [2]. This fits in line with our FourSquare Check-in dataset. The census dataset maps 2167 individual census tracts, which are geographic divisions that are more fine-grained than zip-codes, to demographics like income, total population, and unemployment rate. For our approach, we are interested in *income*, but other demographics could be feasibly used for analysis.

2.2 Preprocessing

In order to understand check-in data from the FourSquare dataset in relation to census tracts, we needed to map each of the latitude/longitude pairs in each check-in to a census tract. To do this, we used the FCC's Census Block Conversions API [3], which converted each coordinate to a census block code. Census blocks are the smaller regional units used by the U.S. census which are then grouped into census tracts. We truncated the last 4 digits of the block code to convert it into our desired census tract. For all 227k check-in samples, this process took approximately 22 hours on a CPU. After this step of preprocessing, we could query all check-ins within a given census tract.

In our experiments, we modeled wealth via income. To do this, we took the `income` column of the NYC census dataset. Of the 2167 census tracts, 66 of them had `null` values for income, so we filtered the dataset to the remaining 2101 examples. Next, we bucketed our income labels into quartiles, such that each bucket would contain 25% of the data points. This way, we could separate our buckets into general ranges of wealth while maintaining class balance.

2.3 Feature Extraction

We manually constructed features (Table 1) that could potentially translate into predictive power. For each potential feature, we counted the number of check-ins that corresponded to each of our features, and normalized based on the number of check-ins in that feature category (feature `b` in Table 1). We show visualizations of some potential features compared to income in Fig. 1.

Table 1: Features extracted from Check-in Data for each Census Tract

a	→	num_total_checkins
b	→	num_checkins_per_category (e.g. mexican_restaurant, medical_center)
c	→	num_checkins_per_weekday (e.g. Mon, Tue)
d	→	num_checkins_per_timebucket (e.g. 8am_to_12pm)
e	→	num_checkins_per_category_per_weekday
f	→	num_checkins_per_category_per_timebucket
g	→	num_checkins_per_category_per_weekday_per_timebucket

These features came from intuition that there might be relationships between the types of businesses that individuals visit and their wealth. For example, the normalized count of each of the 251 venue categories served as signals for our model. More specifically, for census 360050000200, one feature accounting for normalized Mexican restaurant check-ins would have been represented as:

$$\frac{\texttt{mexican\_restaurant\_count}}{\texttt{total\_checkin\_count}}|\texttt{census\_tract} = 360050000200$$

In a similar way, we computed the normalized features for days of week (feature c in Table 1) and 4-hour time buckets (d in Table 1). We also engineered features for combinations of the previous features. For example, we found that there might exist an association between income and coffee shop visit between 8am and noon. As a result, these composite features (e through f in Table 1) captured more complex relationships about the habits of individuals and the businesses they frequented.

Table 2: Feature Selection Results with Naive Bayes Model

		Statistics	
Features* included	Dimension (size of n)	Cross Val. Train	Cross Val. Dev
{a,b}	251	0.4282	0.3182
{a,b,c}	258	0.4521	0.3251
{a,b,c,d}	**264**	**0.4521**	**0.3491**
{a,b,c,d,e}	1615	0.5287	0.3057
{a,b,c,d,e,f}	1657	0.5390	0.3229
{a,b,c,d,e,f,g}	9028	0.5283	0.2857

** as defined in Table 1*

2.4 Feature Selection

In order to choose the features that we found to be most relevant, we evaluated each set of features against our naive bayes baseline, as shown in Table 2. Following our intuition, we found that the more complex, composite features seemed to overfit on the train data during cross-validation, whereas the lack of feature tended to underfit. We ultimately chose feature sets a,b,c,d (based on Table 1) for further evaluation, based on the best cross validation development accuracy, as indicated in Table 2.

2.5 Model Evaluation

Finally, we used these features to evaluate our results on a number of different models, which will be discussed in section 3.1. Because we have a small dataset ($m = 2101$), we needed to overcome overfitting by applying a number of regularization methods to each of the models.

With 4 equally-balanced classes, we would expect a model operating on chance to achieve an accuracy of 25%. To evaluate our models, we split the test set into 6 separate folds. We then performed

Table 3: Income Classification Results

Model	Statistics Cross Val. Train	 Cross Val. Dev	 Test
Naive Bayes (NB)*	0.4311	0.3223	0.2906
Logistic Regression (LR)	0.4231	0.3269	0.3333
Support Vector Machine (SVM)	0.2671	0.2383	0.2165
Quadratic Discriminant Analysis (GDA)	0.6160	0.3229	0.3533
Gradient Boosting Tree (GBT)	**0.6990**	**0.3794**	**0.4103**
Multilayer Perceptron (MLP)	0.5386	0.3326	0.3305

* *baseline*

k-fold cross-validation on the first five folds to tune certain hyperparameters (regularization penalties, depth in decision trees etc.) before finally testing each model on the 6th, held-out test set.

3 Experimental Evaluation

3.1 Models

Baseline: **Naive Bayes (NB)** We use Naive Bayes with the caveat that our data violates the assumption of conditional independence, because multiple features might rely on the same time or category. The Naive Bayes model uses each feature's set of counts as a predictive feature.

Logistic Regression Logistic regression is a simple model that captures weights of features in terms of predictive importance. In this case, however, it fails to capture complex relationships between features.

Support Vector Machine (SVM) The SVM classifier was intended to linearly separate the data, but it performed poorly because our feature space is likely not linearly-separable.

Quadratic Discriminant Analysis (QDA) QDA assumes an independent distribution for each income quartile, but its poor performance shows QDA fails to adequately generalize its learned distribution during evaluation on the test set, likely due to the overlapping influence between features.

Gradient Boosting Trees (GBT) The model learns by building complex, nonlinear decision boundaries for difficult to classify examples. Of all our models, GBT performed the best, likely by selecting and combining small subsets of features via an ensemble of weak prediction models.

Multilayer Perceptron (MLP) We use a 4-layer feed-forward network with hidden layer sizes $(5, 5, 4)$ and a ReLU activation function to attempt to capture complex relationships between our low-level features. While this model performed decently, it didn't beat other models, likely because our input features were limited.

3.2 Discussion

The results for each of these models is shown in Table 3. Gradient boosting trees perform the best, achieving an average 37.94% accuracy on the dev set during cross validation, and a 41.03% accuracy on the test set. This represents a significant improvement over our NB baseline, which only achieved 32.23% accuracy on the dev set and 29.06% on the test set.

The confusion matrix (Fig. 2) of the predicted income quartiles shows that our model performs better when predicting higher income quartiles (Q3 and Q4). This result aligns with our expectations, because the dataset is more feature-rich in high-income areas, as shown in the Fig. 1 heatmaps. Intuitively, people in less-wealthy areas might be less likely to have high smartphone usage, let alone FourSquare usage, which translates to weak signals in our lower-income census tracts.

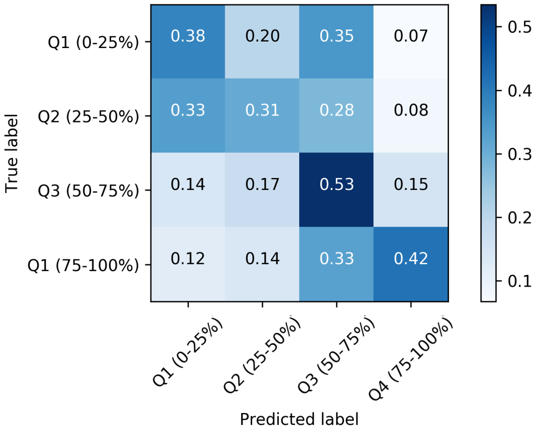


Figure 2: Confusion matrix for income classification based on check-in features.

3.3 Feature Analysis

We seek to understand the most predictive features for each of the four income buckets. We find weights for each feature by fitting a logistic regression model to the dataset and saving the feature coefficients in the model. By sorting these coefficients, we were able to identify the most positive coefficients as corresponding to the most predictive features. We see in Table 4 that the venue categories with high predictive power might match our assumptions about a particular neighborhood. For instance, activity around housing developments and fast food restaurants are associated with the lower income quartiles, while vegan restaurants and casinos appear more often in the wealthier income-brackets.

Table 4: Feature Analysis using Weights from Logistic Regression

Income Quartile	Top Five Most Predictive Features
Q1:(0% - 25%)	num_factory, num_residential_apartment_condo, num_housing_development, num_school, num_synagogue
Q2:(25% - 50%)	num_storage_facility, num_fast_food_restaurant, num_subway, num_automotive_shop, num_deli_bodega
Q3:(50% - 75%)	num_bridal_shop, num_bar, num_home_private, num_tattoo_parlor, num_casino
Q4:(75% - 100%)	num_vegetarian_vegan_restaurant, num_coffee_shop, num_harbor_marina, num_cupcake_shop, num_art_museum,

4 Future Work

While we only focused on predicting income in this work, our feature analysis leads us to believe that we would find similarly interesting results with other demographics, especially because other demographics (e.g. gender, ethnic composition, population size) are often also correlated with income. For example, we might also find new relationships about how certain culturally-themed businesses (e.g. Dim Sum restaurants) might reflect certain characteristics of a neighborhood (e.g. Chinatown). Ultimately, we are excited about the possibilities of our work because in a social sciences context, they might not only confirm or deny beliefs about the demographics surrounding businesses, but also help us discover new relationships.

Acknowledgments

We would like to extend a special thanks to our project mentor, Qijia Jiang, for taking the time to meet with us to refine and validate our idea. We would also like to thank the CS229 course staff and professors for their hard work in facilitating a wonderful course!

Individual Contributions

Vincent handled the data discovery, pre-processing, and extensive feature engineering. We separately explored the different models that we could potentially use. Dan set up the model pipeline/infrastructure and conducted the feature analysis (including the creation of the feature heatmaps). Both team members performed parameter tuning and error analysis for the different types of models.

References

[1] Dingqi Yang, Daqing Zhang, Vincent W. Zheng, Zhiyong Yu. Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs. IEEE Trans. on Systems, Man, and Cybernetics: Systems, (TSMC), 45(1), 129-142, 2015. [PDF]

[2] American Community Survey, *MuonNeutrino* (Kaggle). (2015) New York City Census Data. Retrieved from `https://www.kaggle.com/muonneutrino/new-york-city-census-data`.

[3] Federal Communications Commission. Census Block Conversions API. (2017) Retrieved From `https://www.fcc.gov/general/census-block-conversions-api`.