Gentrification: Causation and Identification

Ramin Ahmari
Stanford University

Soraya Karimi
Stanford University

Kenneth Xu
Stanford University

Matt Mistele
Stanford University

Abstract— **Gentrification is becoming an increasingly divisive and impactful sociological and political issue across developed countries. This paper investigates the application of various machine learning techniques to calculate and predict a gentrification susceptibility index for an area given a satellite image of that area as a possible tool for legislators examining their county's or city's area structure. After comparing different models, a random forest model was applied to the problem with features extracted from the satellite images leading us to a 0.61 test error for predicting the 6-class label directly and a 0.57 test error rate by summing the predictions for its 6 component factors.**

I. INTRODUCTION

Gentrification is described as the entry of affluent residents into established urban and socioeconomically disadvantaged districts and neighborhoods, leading to renovation and revival as well as increased property values and the displacement of traditionally low-income families and small businesses. An intricate undergoing that is organic to any city, gentrification can lead to higher median income and job opportunities offering low-income families an increased socioeconomic status, yet it can also cause displacement of families falling below a certain wealth threshold unable to afford the increased rent and everyday costs. This paper and project intend to act as a tool for legislators to identify areas susceptible to gentrification and plan political activities beforehand for mitigation purposes.

To do so, we created an overall gentrification susceptibility index comprised of 6 gentrification factors that we took from Chapple et al. (2009) [1] (percentage of housing units that are 5+ individual units, percentage of occupied housing that's occupied by renters, percentage of workers taking public transportation, median gross rent, percentage of households that are non-family households, percentage of renters paying over 35%) of their income). We also looked at 3 more factors that Chapple et al. did not list as having a significant difference in between the study's gentrifying and non-gentrifying areas, but which are related to gentrification and might be of interest to the same audience: Gini Index, population density, income diversity. We then trained support vector machines, a logistic regression model, and a random forest model on the data set, and we found that the random forest model yielded the best results.

As input to our model, we used satellite imagery using a bounding box to approximate certain zip codes based on latitudes and longitudes. We predicted each factor individually and the gentrification susceptibility index directly from the data. We furthermore summed up the six Chapple et al. (2009) factors we predicted individually to create another prediction of the gentrification susceptibility index and calculated error rates for all predictions.

e-mail: rahmari@stanford.edu
e-mail: skarimi@stanford.edu
e-mail: kenxu95@stanford.edu
e-mail: mmistele@stanford.edu

II. RELATED WORK

Our work was inspired in part by Stanford Professor Ermon's research on training a convolution neural network (CNN) on satellite images to identify regions of poverty in Africa. [2] The challenge of finding relevant and substantial data in Ermon's study was resolved through a multi-step process called transfer learning, which utilizes noisy but easily obtainable images to train the deep learning machine model. After pre-training the CNN on thousands of images, the CNN measured poverty by extracting daytime imagery features to infer variation of nighttime light, given that nighttime lighting can serve as a rough proxy of economic prosperity. Ermon's unique approach to measuring nightlight proved successful, with cross-validated predictions explaining 44-59% of variation in average household wealth across 5 African countries. Comparatively, research analyzing street-view imagery utilized the output of a pre-trained convolution network on an SVM to classify a Beijing area as 'beautiful' or 'ugly' with 75% accuracy, demonstrating how broad scale terrain classification can be effective given the specificity of the dataset.[3]

While imagery-based research on gentrification proved scarce, many researchers took advantage of the abundance of census tract data to produce various measurements of gentrification. For example, Binet (2016) utilized a Markov chain to model socioeconomic state changes between six "region types" found using k-means clustering, one region of which was "gentrifying." Transitions from each of the other five states to this one thus provided an estimate of the probability of gentrification for a given New York region. [4]

Although lacking in spatial input, feature estimates were widely available for Binet's region of interest, again proving how crucial obtaining data is in rigorous and comprehensive machine model training. An alternative, "metadata"-based approach to measure gentrification was conducted by Chapple et al. at Berkeley (2009), using census data to form a gentrification susceptibility indicator index. Chapple trained a multivariate linear regression model on 19 metadata features such as household income, race, number of housing units, etc., ranked based on how well each feature predicted whether an area in the 1990s Bay Area would have undergone gentrification in the following 10 years, to

compile an index of gentrification susceptibility that can be calculated for any region. [1] We adopted 6 susceptibility factors described in this paper that are freely available from the American Community Survey for areas across the country, forming a new index from 0 to 6. We also predicted three additional factors separately that we suspected would be related to gentrification nation-wide, though not in the Bay Area regions of Chapple et al.'s study, for a more comprehensive picture of gentrification.

An interesting approach to measuring gentrification in Milan was conducted by using telephone interviews as the dataset. By characterizing the population of newly moved in residents and training a self-organizing neural network map on the results, Diappi et al. (2013) mapped patterns and trajectories in the driving forces behind gentrifiers, such as family needs and housing demands.[5]

Despite the extensive machine learning based research that has been done with satellite imagery and gentrification respectively, no work has been documented that uses spatial data to directly measure gentrification. Previous research has informed our decisions on algorithm structures and feature extraction as well as hyperparameter adjustments.

III. DATASET & FEATURES

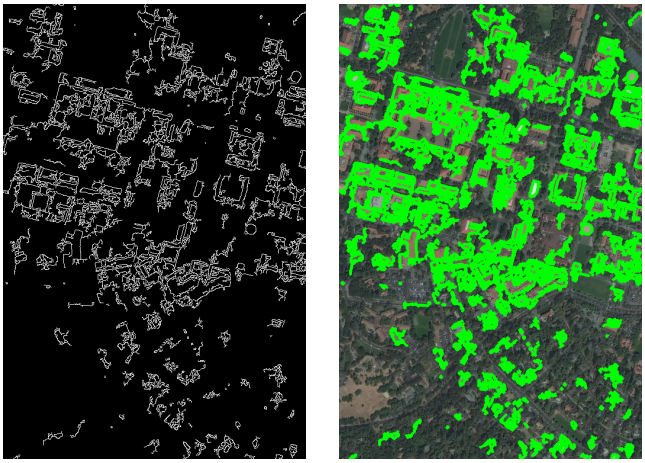
We selected 6479 ZIP codes from across the country using Social Explorer, selecting all the ZIP codes within a reasonable radius from each major US city visible at zoom level 6. We then used the site to download the American Community Survey's 2010-2014 5-year estimates of social and economic statistics ("metadata" for the images) for each ZIP code [6], and we used the CivicSpace US ZIP Code Database to add the latitude and longitude of each ZIP code [7]. For the purposes of matching satellite images to ZIP codes, we approximated the bounding boxes of each ZIP code as squares centered at the given latitude and longitude and with area equal to the area statistic provided in the American Community Survey. From this census data, we were able to calculate the values of 8 factors found by Chapple et al. (2009) to be most correlated with gentrification (at least in the 1990s Bay Area). In the style of Chapple et al, we integrated these factors into a rough "susceptibility to gentrification index", an integer between 0 and 6 indicating the number of factors with respect to which the area was closer to the mean of gentrifying Bay Area areas than non-gentrifying Bay Area areas.

Satellite images were downloaded through DigitalGlobe Recent Imagery API, giving us an expansive and current snapshot of the world's surface. We were able to extract 700 satellite images, each covering a zip code to the nearest 0.3 mile, by calculating the longitude-latitude bounding boxes for each. These images were on the order of 800px x 800px. (For example, the images of Stanford below were 512x768 pixels.) Each satellite image was then labelled with its associated metadata, and 100 were set aside for use as test examples. The other 600 were used for training examples.

A. Feature Extraction

1) Edge Detection: Edges are extracted utilizing Canny Edge Detection. The percentage of edges is used as the feature.

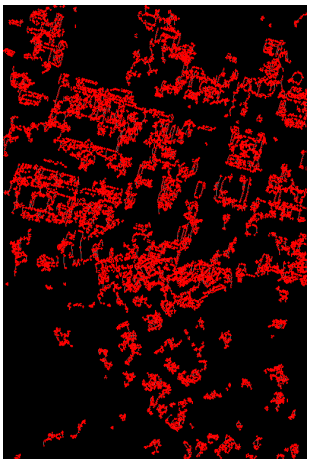
2) Shapes Detection: Shape extraction is performed on the above "edges" image. The total length and size of the shapes, as well as the frequency of different shapes (triangle, circle, etc.) are bucketed and used as features.



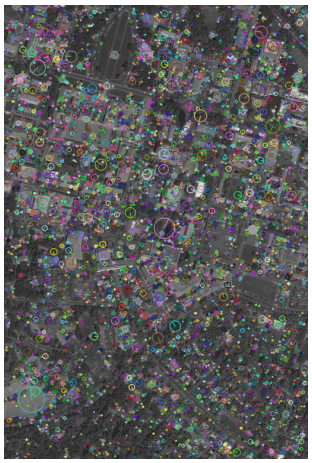
(a) Edge Extraction (b) Shape Extraction

3) Corner Detection: The Harris Corner Extractor algorithm was used on both the original image and the extract edges to obtain points of interest and rough clusters of buildings. The number of corners (normalized) was extracted as a feature.

4) SIFT Extraction: Finds "keypoints" using the SIFT (Scale-Invariant Feature Transform) algorithm, corresponding to corner candidates at different scales, and extracts the "keypoint density" (number of keypoints divided by the size of the image), a 10-bin histogram of the keypoint sizes, and a 10-bin histogram of the keypoint octaves.



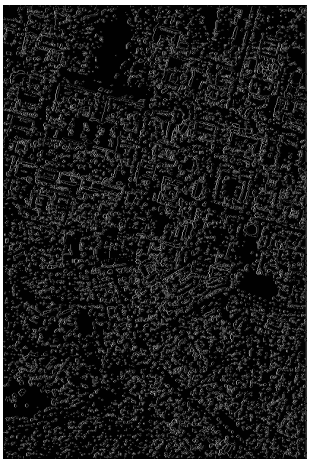
(a) Corner Extraction



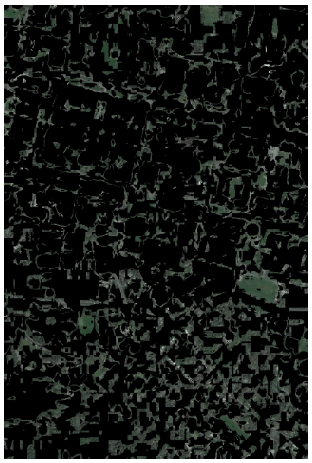
(b) SIFT

5) Texture Extraction: Applies a bilateral filter to the image and then thresholds it to obtain interesting information about heights and compositions of buildings. Unfortunately, we were unable to figure out how to extract that data (and instead used basic black-white percentage).

6) Green Extraction: Applies a mask to the image to remove all the non-green portions. Theoretically, this produces a good indicator for how urbanized a particular region is.



(a) Texture Extraction



(b) Green Extraction

7) Color Histogram: R,G,B colors in the image are bucketed into 256 bins. We break these up into 16 bins - 1 if it is an above-average bin and 0 otherwise.

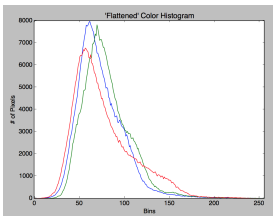


Fig. 4: Color Histogram

IV. METHODS

A. Logistic Regression

Logistic regression is a statistical classification model with a categorical dependent variable. The logistic model estimates the probability that an example is in a given class, such as "above-average on this variable", by using the cumulative logistic distribution to estimate the higher-dimensional probability curve over the range of possible feature vectors. This distribution is calculated with the logistic function

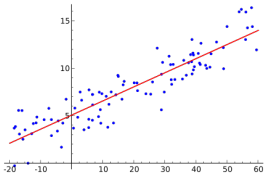


Fig. 5: Logistic Regression
(Source: RaphaelQS, Wikipedia Commons, March 2016)

$$f(x) = \frac{1}{1 + e^{-t}} \tag{1}$$

for which t = a linear function of a single explanatory variable x (such as a dot product of a weights vector and the feature vector x).

We used logistic regression to classify each ZIP code as having an above-average or below-average value of each label (corresponding to values 1 and 0).

Estimations were derived through the ordinary least squares calculation, since the logistic regression model can be expressed as a generalized linear model.

B. Support Vector Machine

Support Vector Machines (SVMs) are machine learning models that estimate a categorical output based on certain features that we identify as a non-probabilistic binary classifier.
Graphically, SVM estimates the output by dividing a set of points with a line in a way that the gap between the points on either side and the line is as large as possible (indicating a clear and evident separation and classification). When handling new data, this line will be evaluated to establish the categorization of the new example data, determining on which side of the gap the new example point fall to formulate its classification.

The classifier of a Support Vector Machine for a p-dimensional vector is defined by the (p-1)-dimensional hyperplane that separates the examples as described above. The optimal hyperplane is defined to be associated with the biggest margin to its closest example point. This hyperplane is denoted as the maximum-margin hyperplane which defines the maximum-margin classifier. It is this maximum-margin hyperplane that a Support Vector Machine establishes.

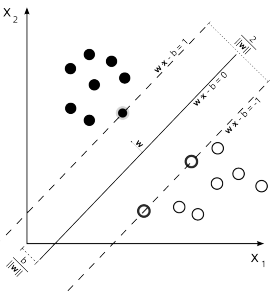


Fig. 6: Maximum Margins / Gaps Created Through Hyperplane
(Source: Peter Buch, Wikipedia Commons, June 2011)

Similarly to our methodology for logistic regression, we implemented an SVM to classify each ZIP code as having an above-average (1) or below average (0) value for each of the labels of interest (population density, Gini index, etc.).

C. Random Forest

Random decision forests are an ensemble learning method that can be used for both regression and classification. Ensemble learning algorithms are models that, in a divide-and-conquer approach, use multiple learning algorithms to boost their prediction accuracy and modeling behavior.

Random decision forests in particular have gained their name by establishing numerous decision trees (a popular choice for data mining purposes as they are invariant under scaling and certain other transformations of the extracted features that are inputted into them) when being trained on the training data. However, each tree is a in itself only a weak learner due to its low bias but high variance, yet together they can lead to a strong learner. This process is represented graphically on Figure 7. Thus, for classification, the mode of the different decision tree is outputted and for regression, the mode of the mean prediction of the different decision trees is outputted.

Achieving the strong learner from the weak learners is being done through Bootstrap Aggregating which in this case works as follows:
For some number of trees T:

1) Sample N cases at random (with replacement). This creates a subset of the data that is usually aimed to be about two thirds of the entire data set.
2) Repeat for each node: m predictor variables are randomly selected from all predictor variables with the value of m being much smaller than the total amount of predictor variables. The predictor variable that achieves the best binary split (binary splitting is a technique for increasing the efficiency of numerical evaluations) on that node splits it.

Algorithm explanation adopted from [8]

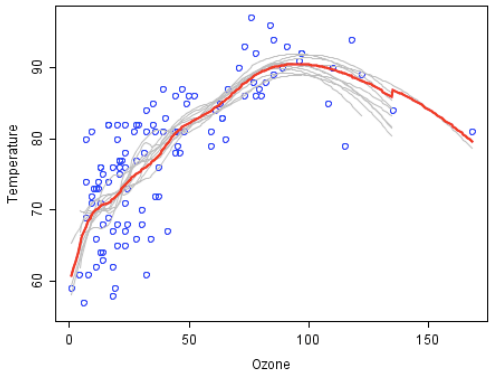


Fig. 7: Creating a Strong Learner (Red) By Aggregating Weak Learners (Gray) Over the Data (Blue)
(Source: Future Perfect at Sunrise, Wikipedia Commons, May 2012)

D. Further Notes

As decision trees trend to overfit their training sets, random decision trees actively correct themselves for this phenomenon by averaging the decision trees, trained on different parts of the same training data. This is to increase the variance of the algorithm as it already has low bias due to the nature of the decision trees.

We split all of our labels into 2 classes: bottom 50% and top 50%. Each label was a particular feature of gentrification (as given by University of Berkeley's 2009 report on Gentrification): population-density, % of renter-occupied housing, Gini index, etc. Thus, we trained our data on each of these features to discern whether our models could accurately identify correlations and causes of these gentrification features from only satellite imagery.

We split our data into 600 training examples and 100 (unchanging) testing examples. We decided to first plot learning curves for our model (averaged over 10 trials). Our baseline model was very simple: edge detection with SVM and logistic regression to predict population density (an aspect with more obvious associations with our extracted features). From these learning curves, we realized that SVM was a terrible learner for this problem, thus striking it from our experiment.

V. RESULTS

The tables below show results when applying either the logistic regression model or our random forest model. The random forest model proved to perform better than the regression model. Interestingly enough, median gross rent's error rate decreased from 42% to 33% when we switched to the random forest model which shows how effective this model can be since making a connection between median gross rent and a simple satellite image is difficult since,

compared to an architectural feature such as the number of housing units, the connection to the satellite image is quite weak.

Our overall gentrification susceptibility index was predicted in two ways: directly from the data and by summing up the predicted the individual factors that make up the gentrification susceptibility index. We saw a small improvement in error rates when using the summation method. This must stem from the fact that a multi-class classification problem is just inherently more difficult than a binary classification and the summation consequently performs slightly better.

Furthermore, it is observable that across all metrics, the random forest outperformed the logistic regression, especially when it comes to the training data. We initially only ran the support vector machines and logistic regression model and were alarmed by the high training error which showed high bias and thus under-fitting. The random forest model addresses high bias issues with its inherent decision tree structure and consequently we constructed and applied that model to yield the far better results.

Running a logistic model on the individual gentrification factors (split into top 50% and bottom 50%), we obtain:

Gentrification Factor	Training Error	Test Error
% housing that are 5+ units	0.31	0.39
% renter-occupied housing	0.28	0.33
% workers taking public transport	0.27	0.32
Median Gross Rent	0.28	0.42
% Non-Family Households	0.30	0.28
% renters paying over 35%	0.31	0.42
Income Diversity	0.29	0.34
Gini Index	0.31	0.38
Population Density	0.26	0.23

Running a random forest classifier on the individual gentrification factors, with hyperparameters: number of trees = 300, maximum number of features = 30, maximum depth = 10, and minimum samples per leaf = 2 chosen to reduce overfitting, we obtain:

Gentrification Factor	Training Error	Test Error
% housing that are 5+ units	0.00	0.33
% renter-occupied housing	0.00	0.29
% workers taking public transport	0.00	0.27
Median Gross Rent	0.00	0.33
% Non-Family Households	0.00	0.28
% renters paying over 35%	0.00	0.42
Income Diversity	0.00	0.30
Gini Index	0.01	0.41
Population Density	0.00	0.24

Finally, predicting the gentrification susceptibility index (0-6) with logistic regression:

	Training Error	Test Error
Directly predict the index	0.50	0.61
Summation over the individual thresholds	0.18	0.60

We get the following confusion matrix:

0	7	0	0	0	0	0
0	36	6	0	0	0	0
0	12	5	3	0	0	1
0	8	4	2	0	2	0
0	6	2	2	0	0	0
0	4	0	0	0	0	0
0	0	0	0	0	0	0

And random forests:

	Training Error	Test Error
Directly predict the index	0.02	0.53
Summation over the individual thresholds	0.02	0.57

From which, we get the following confusion matrix:

0	7	0	0	0	0	0
0	39	3	0	0	0	0
0	16	4	1	0	0	0
0	6	5	4	0	1	0
0	9	1	0	0	0	0
0	3	1	0	0	0	0
0	0	0	0	0	0	0

VI. CONCLUSION AND FUTURE WORK

Considering that the gentrification susceptibility index error rate should be seen in relation to a 6-class classification, we performed quite well on the prediction of individual futures and well on the prediction of the overall gentrification susceptibility index.

One aspect to consider is that machine learning algorithms due better with more data and, as we were only able to use 700 satellite images due to our restricted time frame and data access, error rates can be much improved with more input data. If more time were available, and as future prospects for this project, we would aim to gather and incorporate more data into our model and furthermore expand our input dimensionality by adding meta data for prediction purposes rather than focusing on satellite images only as that would allow for more rigorous prediction since images hold only so much information. Furthermore, we could try to generalize and find access to data relating to the other 13 features mentioned in Chapple et al. (2009)[1] and also include those to make our model more realistic.

Lastly, a neural network, coupled with more images, would be able to identify more useful features that potentially would be very useful for our classifier.

REFERENCES

[1] Karen Chapple et al. Mapping susceptibility to gentrification: The early warning toolkit. page 28, August 2009.

[2] Neal Jean, Marshall Burke, Michael Xie, W. Matthew Davis, David B. Lobell, and Stefano Ermon. Combining satellite imagery and machine learning to predict poverty. *Science*, 353:790–794, 2016.

[3] Lun Liu, Hui Wang, and Chunyang Wu. A machine learning method for the large-scale evaluation of urban visual environment. *CoRR*, abs/1608.03396, 2016.

[4] Emily Binet Royall. Towards an epidemiology of gentrification: Modeling urban change as a probalistic process using k-means clustering and markov models. page 24, 2016.

[5] Lidia Diappi, Paola Bolchi, and Luca Gaeta. Gentrification without exclusion? a som neural network investigation on the isola district in milan. page 22, 2013.
[6] American Community Survey 5-Year Estimates. Comprehensive tables. 2014.
[7] Schuyler Erle. Civicspace us zip code database. 2004.
[8] Dan Benyamin. Facebook ad optimization, facebook ad targeting, facebook audience optimization, facebook audience prediction, tech. 2016.