Online Appendix For:

Estimating Heterogeneous Consumer Preferences for Restaurants and Travel Time Using Mobile Location Data

By Susan Athey, David Blei, Robert Donnelly, Francisco Ruiz and Tobias Schmidt

Draft: January 22, 2018

This Online Appendix begins by providing details of the data and dataset creation. Next we provide estimation details. Then, we provide a variety of results about goodness of fit and our model estimates, including summaries of estimated sensitivity to distance broken out by restaurant category and other characteristics. Finally, we provide details of our analyses of restaurant openings and closings, as well as counterfactual analyses about the ideal locations of restaurants of different categories.

Online Appendix

A1. Data Description

Our dataset is constructed using data from SafeGraph, a company which aggregates locational information from anonymous consumers who have opted in to

* Athey: Stanford University, 655 Knight Way, Stanford, CA 94305, athey@stanford.edu. Blei: Columbia University, Department of Computer Science, New York, NY, 10027, david.blei@columbia.edu. Donnelly: Stanford University, 655 Knight Way, Stanford, CA 94305, rodonn@stanford.edu. Ruiz: Columbia University, Department of Computer Science, New York, NY, 10027, fr2392@columbia.edu, and University of Cambridge, Department of Engineering, Cambridge CB2 1PZ, UK. Schmidt: Stanford University, 655 Knight Way, Stanford, CA 94305, tobiass@stanford.edu. The authors are listed in alphabetical order. We are grateful to SafeGraph and Yelp for providing the data, and to Paula Gablenz, Renee Reynolds, Tony Fan, and Arjun Parthipan for exceptional research assistance. We acknowledge generous financial support from Microsoft Corporation, the Sloan Foundation, the Cyber Initiative at Stanford, and the Office of Naval Research. Ruiz is supported by the EU H2020 programme (Marie Skłodowska-Curie grant agreement 706760).

sharing their location through mobile applications. The data consists of "pings" from consumer phones; each observation includes a unique device id that we associate with a single consumer; the time and date of the ping; and the latitude and longitude and horizontal accuracy of the ping, all for smartphones in use during the sample period from January through October 2017.

Our second data source is Yelp. From Yelp, we obtained a list of restaurants, locations, ratings, price ranges, and categories, and we infer dates of openings and closings from the dates on which consumers created a listing on Yelp or marked a location as closed, respectively.

A2. Dataset Creation and Sample Selection

Our area of interest is the corridor from South San Francisco to South San José around I-101 and I-280. We start with a rough bounding box around the area, find all incorporated cities whose area intersects the bounding box and then remove Fremont, Milpitas, Hayward, Pescadero, Loma Mar, La Honda, Pacifica, Montara, Moss Beach, El Granada, Half Moon Bay, Lexington Hills and Colma from the set because they are too far from the corridor.

This leaves us with the following 41 cities: Los Gatos, Saratoga, Campbell, Cupertino, Los Altos Hills, Monte Sereno, Palo Alto, San José, San Bruno, Atherton, Brisbane, East Palo Alto, Foster City, Hillsborough, Millbrae, Menlo Park, San Mateo, Portola Valley, Sunnyvale, Mountain View, Los Altos, Santa Clara, Belmont, Burlingame, Daly City, San Carlos, South San Francisco, Woodside, Redwood City, Alum Rock, Burbank, Cambrian Park, East Foothills, Emerald Lake Hills, Fruitdale, Highlands-Baywood Park, Ladera, Loyola, North Fair Oaks, Stanford and West Menlo Park.

We then take the shapefiles for these cities as provided by the Census Bureau and find the set of rectangular regions known as geohash5s[1] that cover their union.

[1]Geohashes are a system in which the earth is gridded into a set of successively finer set of rectangles, which are then labelled with alphanumeric strings. These strings can then be used to describe geographic information in databases in a form that is easier to work with than latitudes and longitudes. At its

This is our area of interest and is shown in Figure A.A2.

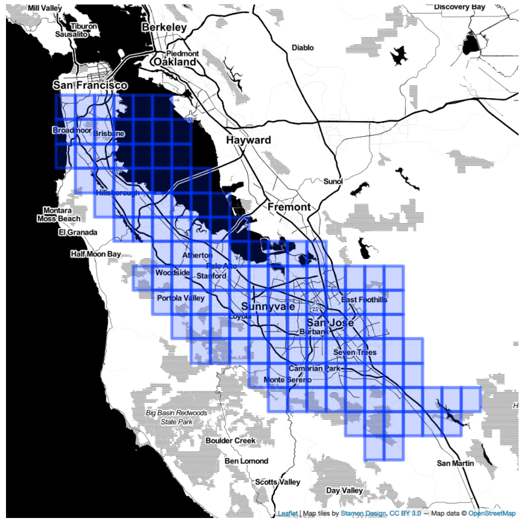


Figure A1. Geographical Region Considered

To construct our *user base* we only consider movement pings emitted on weekdays. We define an active week to be one during which a user emits at least one such ping. The *user base* includes users who meet the following criteria during our sample period, January to October 2017:

- Have an approximate inferred home location as provided by SafeGraph
- Are "active" (defined as having at least 12 — not necessarily consecutive — active weeks)
- Have at least 10 pings in the area of interest on average in active weeks
- 80 percent of pings during hours of 9 — 11:15 a.m. are in the area of interest
- 60 percent of pings during hours of 9 — 11:15 a.m. are in their "broad morning location" where "broad morning location" is at the geohash6 level (a rectangle of roughly 0.75 miles $\times$ 0.4 miles).

coarsest, the geohash1 level, the earth is divided into 32 rectangles whose edges are roughly 3000 miles long. Each geohash1 is then in turn divided into 32 rectangles that are about 800 miles across. The finest geohash resolution used in this paper, geohash8, corresponds to rectangles of size 125 $\times$ 60 feet. See `http://www.geohash.org/` for further details.

- 40 percent of pings during hours of 9 — 11:15 a.m. are in their "narrow morning location" where "narrow morning location" is at the geohash7 level (a square with edge length of roughly 500 feet).
- Have their "broad morning location" in the area of interest

These restrictions give us 32,581 users, which we refer to as our "user base." We then consider the set of restaurants. We begin with the set of restaurants known to Yelp in the San Francisco Bay Area, which we reduce through the following restrictions:

- Locations are in the area of interest
- Locations belong not just to the category "food" but also belong to certain sub-categories (manually) selected from Yelp's list (`https://www.yelp.com/developers/documentation/v2/category_list`): thai, soup, sandwiches, juicebars, chinese, tradamerican, newamerican, bars, breweries, korean, mexican, pizza, coffee, asianfusion, indpak, delis, japanese, pubs, italian, greek, sportsbars, hotdog, burgers, donuts, bagels, spanish, basque, chicken_wings, seafood, mediterranean, portuguese, breakfast_brunch, sushi, taiwanese, hotdogs, mideastern, moroccan, pakistani, vegetarian, vietnamese, kosher, diners, cheese, cuban, latin, french, irish, steak, bbq, vegan, caribbean, brazilian, dimsum, soulfood, cheesesteaks, tapas, german, buffets, fishnchips, delicatessen, tex-mex, wine_bars, african, gastropubs, ethiopian, peruvian, singaporean, malaysian, cajun, cambodian, cafes, halal, raw_food, foodstands, filipino, british, southern, turkish, hungarian, creperies, tapasmallplates, russian, polish, afghani, argentine, belgian, fondue, brasseries, himalayan, persian, indonesian, modern_european, kebab, irish_pubs, mongolian, burmese, hawaiian, cocktailbars, bistros, scandinavian, ukrainian, lebanese, canteen, austrian, scottish, beergarden, arabian, sicilian, comfortfood, beergardens, poutineries, wraps, salad, cantonese, chickenshop, szechuan, puertorican, teppanyaki, dancerestaurants, tuscan, senegalese, rotisserie_chicken, salvado-

ran, izakaya, czechslovakian, colombian, laos, coffeeshops, beerbar, arroceria_paella, hotpot, catalan, laotian, food_court, trinidadian, sardinian, cafeteria, bangladeshi, venezuelan, haitian, dominican, streetvendors, shanghainese, iberian, gelato, ramen, meatballs, armenian, slovakian, czech, falafel, japacurry, tacos, donburi, easternmexican, pueblan, uzbek, sakebars, srilankan, empanadas, syrian, cideries, waffles, nicaraguan, poke, noodles, newmexican, panasian, acaibowls, honduran, guamanian, brewpubs.[2]

This yields a list of locations far too broad. We thus refine the resulting set of locations by removing:

- The coffee and tea chains Starbucks, Peet's and Philz Coffee
- All locations whose name matches the regular expression `(coffee|tea)` but whose name does not start with "coffee"
- All locations whose name matches the regular expression `(donut|doughnut)` but does not contain "bagel"
- All locations whose name matches the regular expression `food court`
- All locations whose name matches the regular expression `mall`
- All locations whose name matches the regular expression `market`
- All locations whose name matches the regular expression `supermarket`
- All locations whose name matches the regular expression `shopping center`
- All locations whose name matches the regular expression `(yogurt|ice cream|dessert)`
- All locations whose name matches the regular expression `cater` but does not match the regular expression `(and|&)` (this is to keep places like "Catering and Cafe" in the sample)

[2]Locations can belong to several categories. The location will be included if any categories match.

- All locations whose name matches the regular expression `truck` and who do not have a street address (these are likely to be food trucks that move around)
- A number of "false positives" manually by name (commonly these are grocery stores, festivals or farmers' markets)
- A number of cafeterias at prominent Bay Area tech companies like Google, VMWare and Oracle

Finally, we review the list of locations that would be removed under these rules and save a few handsful of locations from removal manually.

Applying these restrictions leaves us with 6,819 locations. As a last step we de-duplicate on geohash8. Some locations are so close together that given our matching method we cannot tell them apart and need to decide which of potentially several locations in a geohash8 we want to assign a visit to. In 4,577 cases there is a unique restaurant in the geohash8, while 687 have two, with the remainder having three or more. We de-duplicate using the first restaurant in alphabetical order, leaving us with 5,555 locations. (One reason to remove San Francisco from the sample is that higher density areas have more duplication.) The resulting restaurants are visualized in Figure A.A2.

Next, we define a "visit" to a restaurant. For each user, each restaurant and each day we count the number of pings in the restaurant's geohash8 as well as its immediately adjacent geohash8s as well as the dwelltime, defined as the difference between the earliest and the latest ping seen at the loction during lunch hour. Call any such match a "visit candidate". To get from visit candidates to visits, we impose the requirement that there be at least 2 pings in one of the location's geohash8s and that the dwelltime be at least 3 minutes. We also require that the visit be to a location that has no overlap with either the person's home geohash7 or the geohash7 we have identified as the person's narrow morning location so as to reduce the possibility of mis-identifying people living near a location or working at the location as visiting the location. In cases where a sequence of

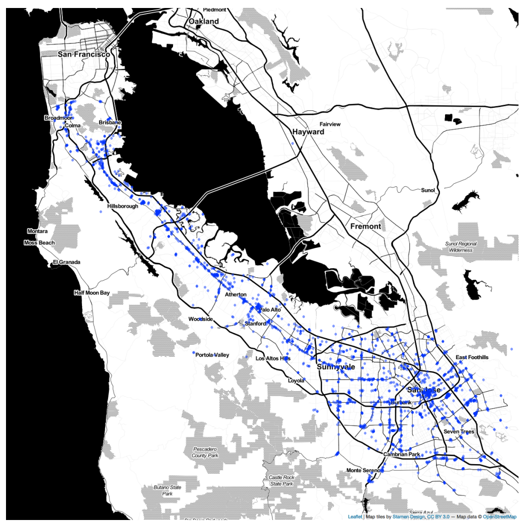


Figure A2. Included Restaurants

pings satisfying these criteria falls into the geohash8s of multiple locations we attribute the visit to the locations for which the dwelltime is longest.

To put together our estimation dataset, we restrict the above visits to a set of users and restaurants we see sufficiently often. We require first that each user have at least 3 visits during the sample period, that each location have at least one visit by someone in the *user base* per week on average, or at least five visits overall (from users overall, not just those in our *user base*). This leaves us with 106,889 lunch visits by 9,188 users to 4,924 locations.

Table A1 provides summary statistics on the users and restaurants included in the dataset.

Table A1—Summary Statistics.

User-Level Statistics					
Variable (Per User)	Mean	25%	50%	75%	% Missing
Total Visits	11.63	4.00	7.00	13.00	—
Distinct Visited Rest.	7.25	3.00	5.00	9.00	—
Distinct Visited Categories	11.60	6.00	10.00	15.00	—
Median Distance (mi.)	3.06	0.89	1.86	3.79	—
Weekly Visits	0.39	0.15	0.25	0.47	—
Weeks Active	31.14	22.00	33.00	41.00	—
Mean Rating of Visited Rest.	3.29	3.00	3.33	3.61	1
Mean Price Range of Visited Rest.	1.55	1.33	1.53	1.75	0.6
Restaurant-Level Statistics					
Variable (Per Restaurant)	Mean	25%	50%	75%	% Missing
Distinct Visitors	13.53	5.00	10.00	19.00	—
Median Distance (mi.)	2.39	0.93	1.72	2.94	—
Weeks Open	42.17	44.00	44.00	44.00	—
Weekly Visits (Opens)	0.54	0.17	0.37	0.72	—
Weekly Visits (Always Open)	0.52	0.16	0.34	0.68	—
Weekly Visits (Closes)	0.53	0.15	0.34	0.67	—
Price Range	1.56	1.00	2.00	2.00	10.66
Rating	3.38	2.89	3.53	4.00	14.52

We also use data from Yelp to infer the dates of restaurant openings and closings. We use the following heuristic: the opening is the date on which a listing was added to the Yelp database, while the closing date is the date on which a restaurant is marked by a member as closed. Figure A.A2 shows the openings and closings throughout the sample period. We focus on openings and closings of restaurants that are considered by users whose morning location is within 3 miles of the opening/closing restaurant and who collectively take at least 500 lunch visits both before and after the change in status.

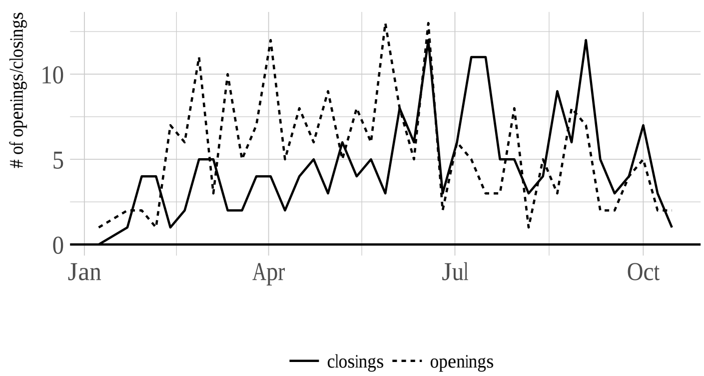


Figure A3. Restaurant Openings and Closings by Week

Distance

As our measure of distance between a user's narrow morning location and each of the items in her choice set we use the simple straight-line distance (taking into account the earth's curvature). After calculating these distances we cull all alternatives that are further than 20 miles away from the choice set.

Item covariates

The following restaurant covariates (or subsets thereof) are used in the estimation of both the MNL and the TTFM:

- `rating_in_sample`: the average rating awarded during the sample period Jan – Oct 2017. If missing the value is replaced by the `rating_in_sample` average and another variable, `rating_in_sample_missing` indicates that this replacement has been made
- `N_ratings_in_sample`: the number of ratings that entered the computation of `rating_in_sample`
- `rating_overall`: the average all–time rating. If missing the value is replaced by the `rating_overall` average and another variable, `rating_overall_missing` indicates that this replacement has been made
- `N_ratings_overall`: the number of ratings that entered the computation of `rating_overall`
- `category_mexican` – `category_dancerestaurants`: A number of 0/1 indicator variables for whether an item has the corresponding category associate with it on Yelp
- `pricerange`: categorical variable indicating the restaurant's price category, from `$` to `$$$$`

A3. Estimation Details

To estimate the TTFM model, we build on the approach outlined in the appendix of Ruiz, Athey and Blei (2017), and indeed we use the same code base, since when we ignore the observable attributes of items, our model is a special case of Ruiz, Athey and Blei. Ruiz, Athey and Blei considers a more complex setting where shoppers consider bundles of items. When restricted to the choice of a single item, the model is identical to TTFM replacing price with distance for TTFM. However, we treat observable characteristics differently in TTFM than Ruiz, Athey and Blei. In the latter, observables enter the consumer's mean utility directly, while in TTFM we incorporate observables by allowing them to shift the mean of the prior distribution of latent restaurant characteristics in a hierarchical model.

We assume that one quarter of latent variables are affected by restaurant price range, one quarter are affected by restaurant categories, one quarter are affected by star ratings, and for one quarter of the latent variables there are no observables shifting the prior.

The TTFM model defines a parameterized utility for each customer and restaurant,

$$U_{uit} = \underbrace{\lambda_i}_{\text{popularity}} + \underbrace{\theta_u^\top \alpha_i}_{\text{customer preferences}} - \underbrace{\gamma_u^\top \beta_i \cdot \log(d_{uit})}_{\text{distance effect}} + \underbrace{\mu_i^\top \delta_{w_{ut}}}_{\text{time-varying effect}} + \underbrace{\epsilon_{uit}}_{\text{noise}},$$

where U_{uit} denotes the utility for the t-th visit of customer u to restaurant i. This expression defines the utility as a function of latent variables which capture restaurant popularity, customer preferences, distance sensitivity, and time-varying effects (e.g., for holidays). All these factors are important because they shape the probabilities for each choice. Below we describe the latent variables in detail.

Restaurant popularity. The term λ_i is an intercept that captures overall (time-invariant) popularity for each restaurant i. Popular restaurant will have higher values of λ_i, which increases their choice probabilities.

Customer preferences. Each customer u has her own preferences, which we wish to infer from the data. We represent the customer preferences with a k_1-vector θ_u for each customer. Similarly, we represent the restaurant latent attributes with a vector α_i of the same length. For each choice, the inner product $\theta_u^\top \alpha_i$ represents how aligned the preferences of customer u and the attributes of restaurant i are. This term increases the utility (and consequently, the probability) of the types of restaurants that the customer tends to prefer.

Distance effects. We next describe how we model the effect of the distance from the customer's morning location to each restaurant. We posit that each customer u has an individualized distance sensitivity for each restaurant i, which is factorized as $\gamma_u^\top \beta_i$, where latent vectors γ_u and β_i have length k_2. Using a matrix factorization approach allows us to decompose the customer/restaurant distance sensitivity matrix into per-customer latent vectors γ_u and per-restaurant latent vectors β_i, both of length k_2, therefore reducing the number of latent variables in the model. Thus, the inner product $\gamma_u^\top \beta_i$ indicates the distance sensitivity, which affects the utility through the term $-\gamma_u^\top \beta_i \cdot \log(d_{uit})$. We place a minus sign in front of the distance effect terms to indicate that the utility decreases with distance.

Time-varying effects. Taking into account time-varying effects allows us to explicitly model how the utilities of restaurants vary with the seasons or as a consequence of holidays. Towards that end we introduce the latent vectors μ_i and δ_w of length $k_3 = 5$. For each restaurant i and calendar week w, the inner product $\mu_i^\top \delta_w$ captures the variation of the utility for that restaurant in that specific week. Note that each trip t of customer u is associated with its corresponding calendar week, w_{ut}.

Noise terms. We place a Gumbel prior over the error (or noise) terms ϵ_{uit}, which leads to a softmax model. That is, the probability that customer u chooses restaurant i in the t-th visit is

$$p(y_{ut} = i) \propto \exp\left\{\lambda_i + \theta_u^\top \alpha_i - \gamma_u^\top \beta_i \cdot \log(d_{uit}) + \mu_i^\top \delta_{w_{ut}}\right\},$$

where y_{ut} denotes the choice.

Hierarchical prior. The resulting TTFM model is similar to the Shopper model (Ruiz, Athey and Blei, 2017), which is a model of market basket data. The TTFM is simpler because it does not consider bundles of products, i.e., we restrict the choices to one restaurant at a time, and thus we do not need to include additional restaurant interaction effects.

A key difference between Shopper and the TTFM is how we deal with low-frequency restaurants. To better capture the latent properties of low-frequency restaurants, we make use of observed restaurant attributes. In particular, we develop a hierarchical model to share statistical strength among the latent attribute vectors α_i and β_i.[3] Inspired by Zhao, Du and Buntime (2017), we place a prior that relates the latent attributes with the observed ones. More in detail, let x_i be the vector of observed attributes for restaurant i, which has length k_{obs}. We consider a hierarchical Gaussian prior over the latent attributes α_i and distance coefficients β_i,

$$p(\alpha_i \mid H_\alpha, x_i) = \frac{1}{(2\pi\sigma_\alpha^2)^{k_1/2}} \exp\left\{-\frac{1}{2\sigma_\alpha^2}||\alpha_i - H_\alpha x_i||_2^2\right\},$$
$$p(\beta_i \mid H_\beta, x_i) = \frac{1}{(2\pi\sigma_\beta^2)^{k_2/2}} \exp\left\{-\frac{1}{2\sigma_\beta^2}||\beta_i - H_\beta x_i||_2^2\right\}.$$

Here, we have introduced the latent matrices H_α and H_β, of sizes $k_1 \times k_{\text{obs}}$ and $k_2 \times k_{\text{obs}}$ respectively, which weigh the contribution of each observed attribute on the latent attributes. In this way, the (weighted) observed attributes of restaurant i can shift the prior mean of the latent attributes. By learning the weighting matrices from the data, we can leverage the information from the observed attributes of high-frequency restaurants to estimate the latent attributes of low-frequency restaurants.

To reduce the number of entries of the weighting matrices, we set some blocks of these matrices to zero. In particular, we assume that one quarter of the latent variables is affected by restaurant price range only, one quarter is affected by restaurant categories, one quarter is affected by star ratings, and for the remaining quarter we assume that there are no observables shifting the prior (which is equivalent to independent priors). We found that this combination of independent and hierarchical priors over the latent variables works well in practice.

To complete the model specification, we place an independent Gaussian prior with zero mean over each latent variable in the model, including the weighting

[3]We could also consider a hierarchical model over the time effect vectors μ_i, but these are low-dimensional and factorize a smaller restaurant/week matrix, so for simplicity we assume independent priors over μ_i.

matrices H_α and H_β. We set the prior variance to one for most variables, except for γ_u and β_i, for which the prior variance is 0.1, and for δ_w and μ_i, for which the prior variance is 0.01. We also set the variance hyperparameters $\sigma_\alpha^2 = \sigma_\beta^2 = 1$.

Inference. As in most Bayesian models the exact posterior over the latent variables is not available in closed form. Thus, we must use approximate Bayesian inference. In this work, we approximate the posterior over the latent variables using variational inference.

Variational inference approximates the posterior with a simpler and tractable distribution (Jordan, 1999; Wainwright and Jordan, 2008). Let $\mathcal{H}$ be the vector of all hidden variables in the model, and $q(\mathcal{H})$ the variational distribution that approximates the posterior over $\mathcal{H}$. In variational inference, we specify a parameterized family of distributions $q(\mathcal{H})$, and then we choose the member of this family that is closest to the exact posterior, where closeness is measured in terms of the Kullback-Leibler (KL) divergence. Thus, variational inference casts inference as an optimization problem. Minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO),

$$\mathcal{L} = \mathbb{E}_{q(\mathcal{H})}\left[\log p(y, \mathcal{H}) - \log q(\mathcal{H})\right],$$

where y denotes the observed data and $\mathcal{L} \leq \log p(y)$. Thus, in variational inference we first find the parameters of the approximating distribution that are closer to the exact posterior, and then we use the resulting distribution $q(\mathcal{H})$ as a proxy for the exact posterior, e.g., to approximate the posterior predictive distribution. For a review of variational inference, see Blei, Kucukelbir and McAuliffe (2017).

Following other successful applications of variational inference, we consider mean-field variational inference, in which the variational distribution $q(\mathcal{H})$ factorizes across all latent variables. We use Gaussian variational factors for all the latent variables in the TTFM model, and therefore, we need to maximize the ELBO $\mathcal{L}$ with respect to the mean and variance parameters of these Gaussian distributions. We use gradient-based stochastic optimization (Robbins and Monro, 1951; Blum, 1954; Bottou, Curtis and Nocedal, 2016) to find these parameters. The stochasticity allows us to overcome two issues: the intractability of the expectations and the large size of the dataset.

The first issue is that the expectations that define the ELBO are intractable. To address that, we take advantage of the fact that the gradient $\nabla\mathcal{L}$ itself can be expressed as an expectation, and we form and follow Monte Carlo estimators of the gradient in the optimization procedure. In particular, we use the reparameterization gradient (Kingma and Welling, 2014; Titsias and Lázaro-Gredilla, 2014; Rezende, Mohamed and Wierstra, 2014). The second issue is that the dataset is large. For that, we introduce a second layer of stochasticity in the optimization procedure by subsampling datapoints at each iteration and scaling the gradient estimate accordingly (Hoffman et al., 2013). Both approaches maintain the unbiasedness of the gradient estimator.

A4. Model Tuning and Goodness of Fit

Table A2 summarizes goodness of fit in the training and test sets. Note, the goodness of fit of MNL is fairly similar across training and test sets, while the much more richly parameterized TTFM model has a large gap in performance between training and test. However, performance of TTFM is substantially better in both training and test sets.

Figure A7 shows how well the model matches the actual purchase probabilities by distance. Figures A4, A5 and A6 show goodness of fit broken out by distance from ther user, by user frequency decile, and by restaurant visit decile for the TTFM and MNL models.

TABLE A2—GOODNESS OF FIT OF ALTERNATIVE MODELS

Model	MSE	Log Likelihood	Precision@1	Precision@5	Precision@10
Training Sample					
TTFM	0.00025	-3.59	31.8%	59.4%	70.3%
MNL	0.00031	-6.58	2.8%	10.7%	16.7%
Held-out Test Sample					
TTFM	0.00028	-5.19	20.5%	35.5%	42.2%
MNL	0.00031	-6.55	3.1%	11.4%	17.5%

Note: Precision measures the share of visits in the set of the top {1,5,10} restaurants predicted by the model.

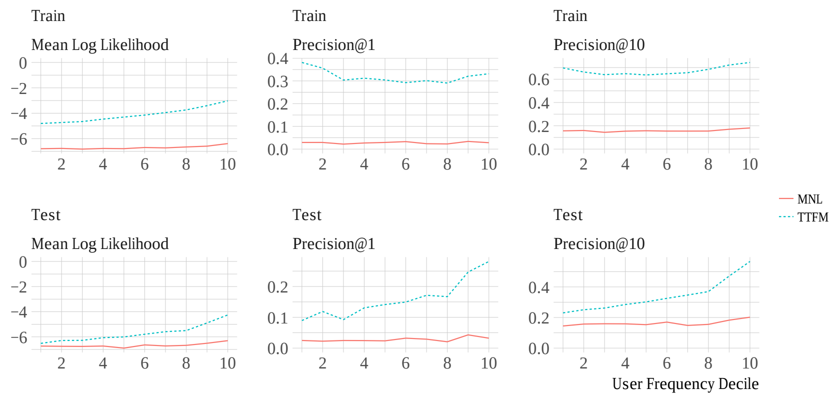


Figure A4. Goodness of Fit Measures by User Decile

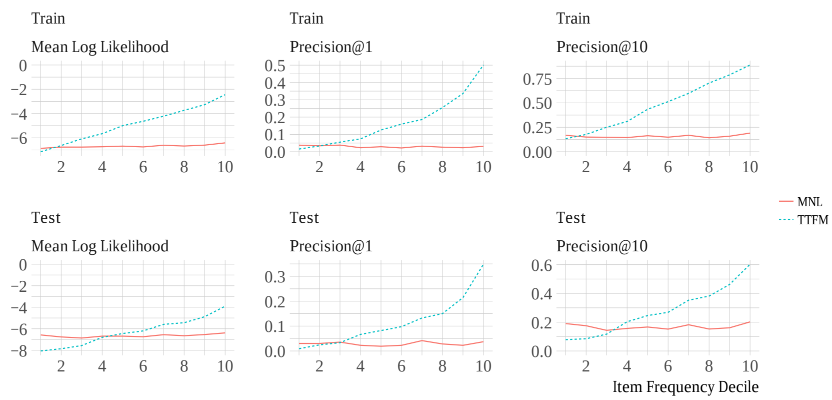


Figure A5. Goodness of Fit Measures by Restaurant Visit Decile

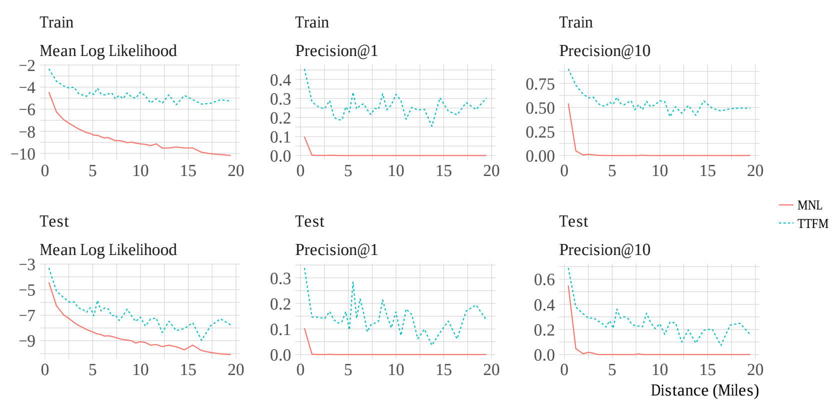


Figure A6. Goodness of Fit Measures by Distance

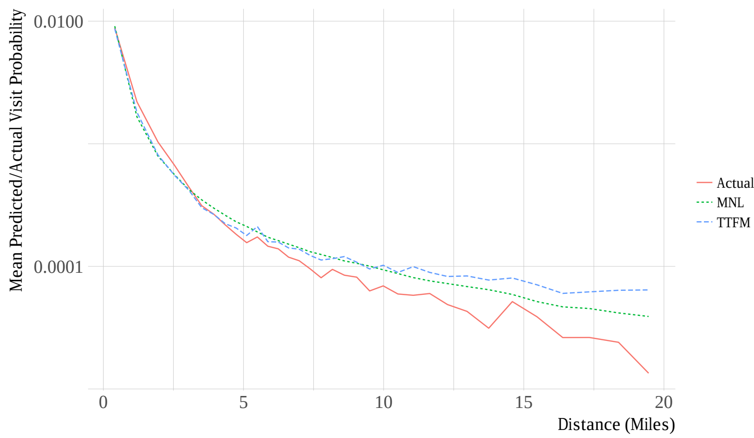


Figure A7. Predicted Versus Actual Shares By Distance

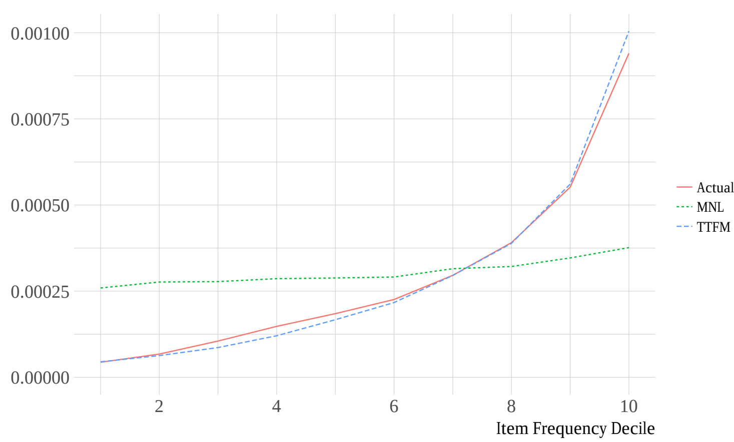


Figure A8. Predicted Versus Actual Shares by Restaurant Visit Decile

A5. Additional Results

Table A3 illustrates how much of the variation in mean item utility (excluding distance) is explained by observable characteristics. All observables combined explain 14 percent of the variation. City and categories each explain $6-7$ percent and lose only a little explanatory power once other variables are accounted for. Star ratings and price range account for 2.8 and 2.3 percent of the variation respectively when considered alone, but only 0.6 percent and 0.4 percent once the other variables are taken into account.

Table A3—Contribution to Mean Item Utility of Observables

Predictors	Variance contribution	Marginal variance contribution
Rating	0.028	0.006
Price range	0.023	0.004
City	0.062	0.053
Categories	0.067	0.046
All	0.140	

Table A4—Distance Elasticities: Summary statistics

Model	Overall		Within-User		Within-Item	
	Mean	SD	SD(Mean)	Mean(SD)	SD(Mean)	Mean(SD)
TTFM	-1.4114	0.6810	0.5992	0.3005	0.2977	0.6003
MNL	-1.4291	0.0033	0.0001	0.0023	0.0002	0.0022

Table A4 gives the means and standard deviations of elasticities in the MNL and TTFM models. Figure A9 plots the distribution of elasticities where the unit of analysis is the restaurant-user pair.

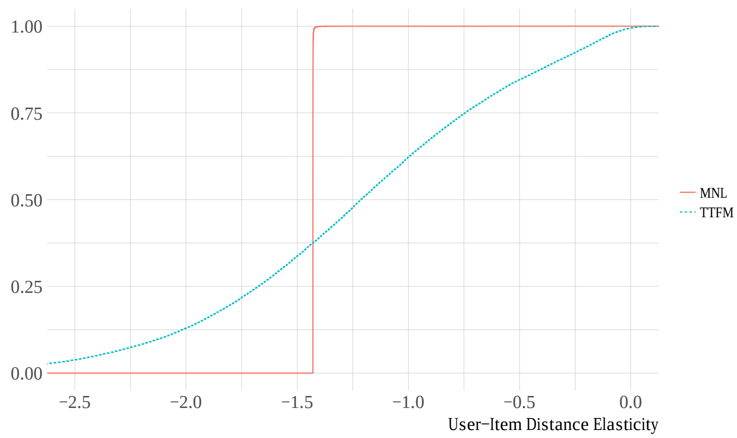


Figure A9. Distribution of Elasticities

Tables A5 and A6 and Figure A10 illustrate how elasticities vary across restaurant types and cities.

Table A5—Average Within-Item Elasticities by Restaurant Characteristics, TTFM model.

Characteristic	Mean	se	25 %	50 %	75 %	N
All restaurants	-1.411	0.0001	-1.585	-1.408	-1.203	4924
Most popular category: Mexican	-1.499	0.0004	-1.664	-1.491	-1.285	694
Most popular category: Sandwiches	-1.435	0.0006	-1.602	-1.441	-1.235	522
Most popular category: Hotdog	-1.403	0.0007	-1.570	-1.390	-1.216	377
Most popular category: Coffee	-1.390	0.0008	-1.563	-1.404	-1.178	365
Most popular category: Bars	-1.370	0.0009	-1.546	-1.362	-1.161	352
Most popular category: Chinese	-1.353	0.0009	-1.517	-1.378	-1.176	350
Most popular category: Japanese	-1.320	0.0011	-1.472	-1.336	-1.140	276
Most popular category: Pizza	-1.497	0.0010	-1.649	-1.481	-1.307	260
Most popular category: Newamerican	-1.323	0.0019	-1.540	-1.351	-1.117	181
Most popular category: Vietnamese	-1.328	0.0020	-1.541	-1.327	-1.155	156
Most popular category: Other	-1.411	0.0002	-1.582	-1.406	-1.189	1391
Price range: 1	-1.446	0.0001	-1.607	-1.435	-1.245	2091
Price range: 2	-1.368	0.0001	-1.542	-1.371	-1.162	2165
Price range: 3	-1.320	0.0026	-1.506	-1.353	-1.108	122
Price range: 4	-1.449	0.0178	-1.664	-1.496	-1.289	21
Price range: missing	-1.474	0.0006	-1.648	-1.455	-1.225	525
Rating, quintile: 1	-1.427	0.0003	-1.605	-1.414	-1.209	842
Rating, quintile: 2	-1.392	0.0003	-1.557	-1.397	-1.187	842
Rating, quintile: 3	-1.364	0.0003	-1.532	-1.366	-1.169	842
Rating, quintile: 4	-1.385	0.0004	-1.571	-1.370	-1.180	842
Rating, quintile: 5	-1.438	0.0003	-1.603	-1.438	-1.250	841
Rating, quintile: missing	-1.475	0.0004	-1.653	-1.464	-1.232	715

Table A6—Average Within-Item Elasticities by City, TTFM model.

Characteristic	Mean	se	25 %	50 %	75 %	N
All restaurants	-1.411	0.0001	-1.585	-1.408	-1.203	4924
City: Daly City	-1.105	0.0019	-1.331	-1.150	-0.959	165
City: Burlingame	-1.119	0.0030	-1.327	-1.194	-1.018	110
City: Millbrae	-1.130	0.0049	-1.418	-1.240	-0.954	80
City: San Bruno	-1.132	0.0035	-1.398	-1.216	-0.987	101
City: South San Francisco	-1.187	0.0021	-1.413	-1.232	-0.999	135
City: San Mateo	-1.243	0.0012	-1.454	-1.284	-1.101	268
City: Foster City	-1.318	0.0070	-1.506	-1.397	-1.163	44
City: San Carlos	-1.321	0.0026	-1.479	-1.350	-1.195	95
City: Palo Alto	-1.330	0.0013	-1.519	-1.342	-1.171	234
City: Brisbane	-1.332	0.0139	-1.455	-1.344	-1.181	15
City: Belmont	-1.334	0.0047	-1.500	-1.374	-1.212	58
City: Redwood City	-1.362	0.0012	-1.530	-1.389	-1.217	214
City: Cupertino	-1.365	0.0018	-1.532	-1.386	-1.174	169
City: East Palo Alto	-1.374	0.0142	-1.521	-1.393	-1.229	13
City: Los Gatos	-1.391	0.0026	-1.583	-1.437	-1.219	106
City: Los Altos	-1.406	0.0043	-1.564	-1.394	-1.236	60
City: Menlo Park	-1.407	0.0031	-1.570	-1.428	-1.287	87
City: Mountain View	-1.422	0.0013	-1.592	-1.429	-1.233	213
City: Santa Clara	-1.442	0.0009	-1.681	-1.456	-1.238	355
City: San Jose	-1.451	0.0002	-1.635	-1.464	-1.278	1858
City: Campbell	-1.482	0.0015	-1.640	-1.493	-1.317	144
City: Saratoga	-1.497	0.0059	-1.628	-1.481	-1.394	40
City: Sunnyvale	-1.501	0.0008	-1.659	-1.513	-1.325	302
City: Stanford	-1.607	0.0062	-1.760	-1.605	-1.482	39

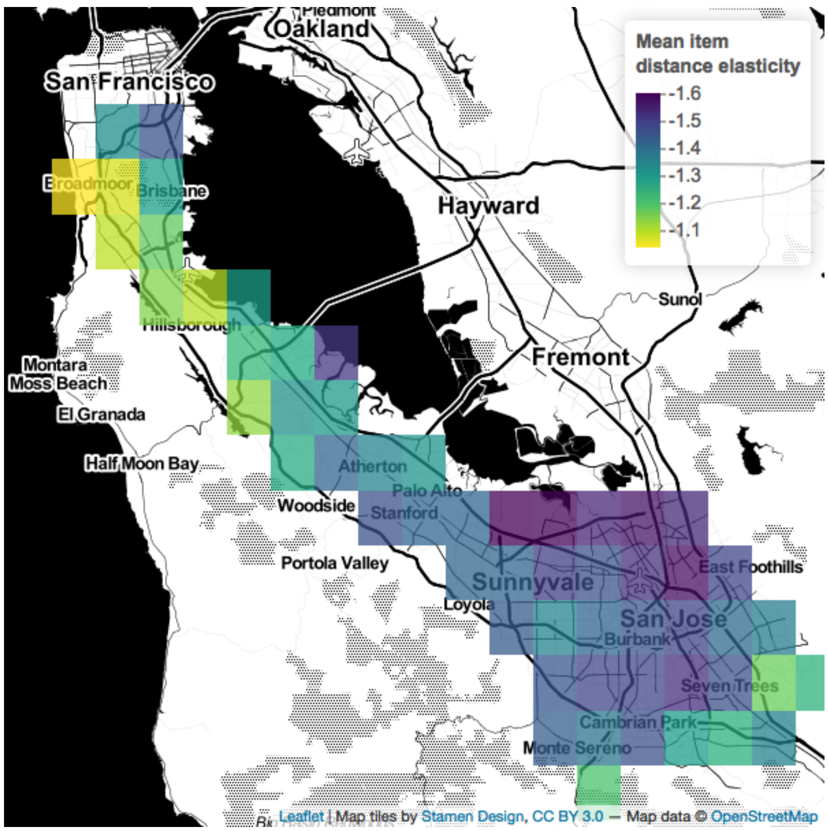


FIGURE A10. AVERAGE WITHIN-ITEM ELASTICITIES BY GEOHASH6, TTFM MODEL.

Tables A7, A8 and A9 illustrate how the model can be used to discover restaurants that are similar in terms of latent characteristics to a target restaurant. Distance between two restaurants, i and i', is calculated as the Euclidean distance between the vectors of latent factors affecting mean utility, α_i and $\alpha_{i'}$. Note that because distance is explicitly accounted for at the user level, we do not expect restaurants with similar latent characteristics to be near one another; rather, they will uncover restaurants that would tend to be visited by the same consumers, if they were (counterfactually) in the same location. We see that indeed, the most similar restaurants to our target restaurants are in quite different geographic locations. Perhaps surprisingly, the category of the similar restaurants is generally different from the target restaurant, suggesting that other factors are important to individuals selecting lunch restaurants.

Table A7—Locations similar (Latent Space) to Curry Up Now in Palo Alto (Indian Fast Food)

Location	City	Category	Distance (Miles)	Latent Distance
Zarzour Kabob & Deli	San Jose	Mideastern	17.2	1.58
Tava Kitchen	Palo Alto	Asian Fusion	0.5	1.62
Pizza Hut	Menlo Park	Pizza	2.8	1.62
Subway	Santa Clara	Sandwiches	11.7	1.62
Rack & Roll BBQ Shack	Redwood City	Seafood	3.8	1.62
Burger King	Redwood City	Burgers	5.2	1.63
Subway	Los Gatos	Sandwiches	19.8	1.64
Pita Salt	Campbell	Street Food	17.0	1.64
Papa John's Pizza	San Jose	Pizza	19.5	1.64
Cutesy Cupcakes	San Jose	Coffee	14.0	1.65

Table A8—Locations similar (Latent Space) to Chipotle Restaurant in Palo Alto (Tacos)

Location	City	Category	Distance (Miles)	Latent Distance
The Van's Restaurant	Belmont	Sandwiches	8.5	1.28
La Viga Seafood Cocina Mexicana	Redwood City	Mexican	4.1	1.31
Three Seasons	Palo Alto	Japanese	0.4	1.32
Cali Spartan Mexican Kitchen	San Jose	Mexican	17.9	1.34
Poor House Bistro	San Jose	Southern	16.8	1.37
McCormick Schmick's Seafood	San Jose	Trad American	17.3	1.38
Taqueria 3 Hermanos	Mountain View	Mexican	6.1	1.38
Peanuts Deluxe Cafe	San Jose	Breakfast	17.4	1.38
Izzy's San Carlos	San Carlos	New American	6.6	1.38
Bibo's Ny Pizza	San Jose	Pizza	18.0	1.39

Table A9—Locations similar (Latent Space) to Go Fish Poke Bar in Palo Alto

Location	City	Category	Distance (Miles)	Latent Distance
Gourmet Franks	Palo Alto	Hotdog	0.03	3.07
Lobster ShackXpress	Palo Alto	Seafood	0.01	3.31
Mayfield Bakery & Cafe	Palo Alto	New American	0.72	3.44
Shalala	Mountain View	Japanese	6.15	3.46
Tin Pot Creamery	Palo Alto	Coffee	0.70	3.47
Mexican Fruit Stand	San Jose	Street Food	18.63	3.60
Leonardo's Italian Deli & Cafe	Millbrae	Coffee	16.50	3.62
Villa Del Sol Argentinian Restaurant	South San Francisco	Latin	19.84	3.63
Bobo Drinks Express	San Jose	Coffee	19.34	3.63
Merlion Restaurant & Bar	Cupertino	Bars	11.81	3.64

Tables A10, A11 and A12 examine restaurants that are similar accounting for all components of utility. Let U_{ui} be the average over dates t that user i visited restaurants of U_{uit}. Distance between two restaurants, i and i', is calculated as the Euclidean distance between the mean utility vectors, $(U_{1i}, .., U_{N_u,i})$ and $(U_{1i'}, .., U_{N_u,i'})$, where N_u is the number of users. Relative to the previous exercise, we see that similar locations are very close geographically, but also still similar in other respects as well. There are many restaurants in close proximity to the selected restaurants, so the list displayed is *n*ot simply the set of closest restaurants.

Table A10—Locations similar (Utility Space) to Curry Up Now in Palo Alto (Indian Fast Food)

Location	City	Category	Distance (Miles)	Latent Distance
Coupa Café	Palo Alto	Coffee	0.09	6.69
Cafe Venetia	Palo Alto	Coffee	0.14	7.54
Jamba Juice	Palo Alto	Juice	0.46	7.72
LYFE Kitchen	Palo Alto	New American	0.17	7.74
Sancho's Taqueria	Palo Alto	Mexican	0.25	7.81
T4	Palo Alto	Coffee	0.18	7.89
Lemonade	Palo Alto	New American	0.19	7.99
Coupa Café	Palo Alto	Coffee	0.28	8.17
Darbar Indian Cuisine	Palo Alto	Indpak	0.27	8.21
Gelataio	Palo Alto	Gelato	0.27	8.23

Table A11—Locations similar (Utility Space) to Chipotle Restaurant in Palo Alto (Mexican)

Location	City	Category	Distance (Miles)	Latent Distance
Bare Bowls	Palo Alto	Juicebars	0.44	6.41
Coconuts Caribbean Restaurant	Palo Alto	Caribbean	0.56	6.63
The Oasis	Menlo Park	Bars	0.36	6.66
Coupa Café	Palo Alto	Coffee	0.48	6.86
Pizza My Heart	Palo Alto	Pizza	0.44	7.07
Fraiche	Palo Alto	Coffee	0.48	7.21
Cafe Del Sol Restaurant	Menlo Park	Mexican	0.86	7.23
MP Mongolian BBQ	Menlo Park	BBQ	0.68	7.34
Bistro Maxine	Palo Alto	Breakfast	0.49	7.85
Koma Sushi Restaurant	Menlo Park	Japanese	0.35	7.88

Table A12—Locations similar (Utility Space) to Go Fish Poke Bar in Palo Alto

Location	City	Category	Distance (Miles)	Latent Distance
Crepevine Restaurant	Palo Alto	New American	0.61	17.96
California Pizza Kitchen	Palo Alto	New American	0.10	18.03
True Food Kitchen	Palo Alto	New American	0.08	19.67
Joya Restaurant	Palo Alto	Mexican	0.58	19.85
Gott's Roadside	Palo Alto	Bars	0.68	20.18
Pressed Juicery	Palo Alto	Juice	0.03	20.37
American Girl	Palo Alto	Trad American	0.10	20.37
Dashi Japanese Restaurant	Menlo Park	Japanese	2.75	20.78
Cafe Bistro	Palo Alto	New American	0.30	20.84
NOLA Restaurant	Palo Alto	Bars	0.54	21.01

A6. Impact of Opening and Closing

Table A13 shows TTFM model predictions for how the opening/closing restaurant's market share is redistributed over other restaurants within certain distances after the restaurant becomes unavailable (i.e. before the opening or after the closing). The results are also illustrated in Figure A11.

Table A13—Share of demand redistributed by distance, TTFM model relative to benchmark

	Distance from opening/closing restaurant (mi.)					
	< 2	2 - 4	4 - 6	6 - 8	8 - 10	> 10
share	51 %	23 %	10 %	6 %	3 %	6 %
cum. share	51 %	74 %	84 %	90 %	94 %	100 %

Figure A12 compares actual changes in market share to predicted changes for competitors of restaurants that opened or closed, grouped by distance from the target restaurant.

A7. Counterfactual Calculations

Our counterfactuals about the match between restaurant characteristics and locations rely on the following type of calculation: how many visits would we predict restaurant i' would receive if it were located in location currently occupied by restaurant i. When we do this, we assume that all characteristics of i', both observed and latent stay the same, except that when we calculate the utility for each consumer for i', we use the location of i when calculating distances. In principle, we can predict the demand i' would receive at any location in the region, however it is easier to have i' replace an existing location i, since this ensures that the chosen location is reasonable (e.g. not in the middle of a forest or a highway).

To calculate demand for i' replacing restaurant i, we calculate new values of the utilities for i' for each user u and session t, which change only due to the new

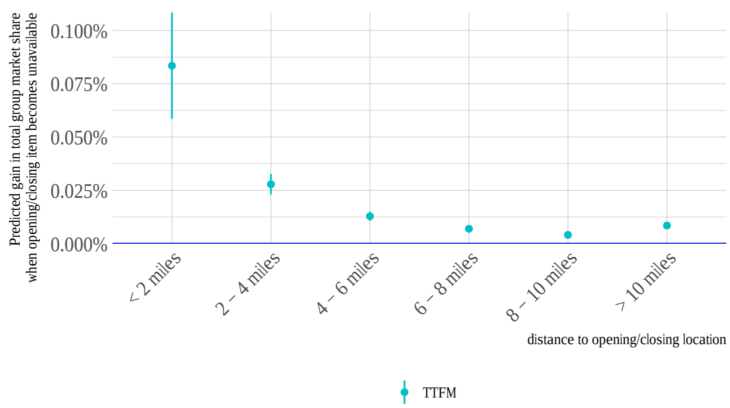


Figure A11. Model Predictions of the Effect of Restaurant Openings and Closings Controlling for Other Changes.

Note: The figure shows the average of the predicted difference in the total market share of each group between the period where the target restaurant is closed and when it is open, minus the difference between the two periods predicted by the model in the counterfactual scenario where the target restaurant is closed in both periods. The user base for the calculated market shares includes all users from the full sample whose morning location is within three miles of the target restaurant and who visit at least one restaurant in both periods. User-item market shares under each regime (target restaurant open and target restaurant closed) are averaged using weights proportional to each user's share of visits in the group to any location during the open period. The bars in the figure show the point estimates plus or minus two times the standard error of the estimate, which is calculated as the standard deviation of the estimates across the different opening or closing events divided by the square root of the number of events.

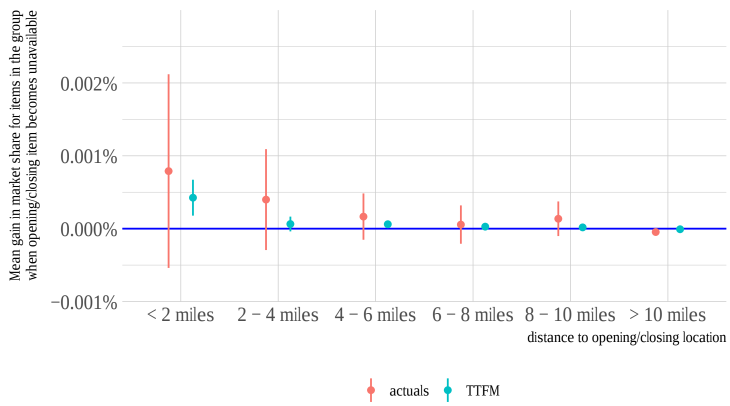


Figure A12. Model Predictions Compared to Actual Outcomes for Restaurant Openings and Closings.

Note: The figure shows the average of the predicted difference in the market share of each restaurant in the group between the period where the target restaurant is closed and when it is open. The user base for the calculated market shares includes all users whose morning location is within three miles of the target restaurant and who visit at least one restaurant in both periods. We consider only restaurants that appear in the consideration sets of these users at least 500 times in both periods. User-item market shares under each regime (target restaurant open and target restaurant closed) are averaged using weights proportional to each user's share of visits in the group to any location during the open period. The bars in the figure show the point estimates plus or minus two times the standard error of the estimate, which is calculated as the standard deviation of the estimates across the different opening or closing events divided by the square root of the number of events.

distances d_{ui} are used instead of the real distances $d_{u,i'}$.

$$U_{uti';i} = U_{uti'} - \gamma_u \beta_{i'} \left(\log(d_{ui}) - \log(d_{ui'})\right)$$

Then we recalculate each user's new choice probabilities in each session, and take the sum across all users and sessions in order to get the new predicted total demand for each restaurant under the counterfactual that i' is located in the location of restaurant i.

$$P(y_{uti';i} = 1) = \frac{\exp(U_{uti';i})}{\exp(U_{uti';i}) + \sum_{l \notin i,i'} \exp(U_{utl})}$$

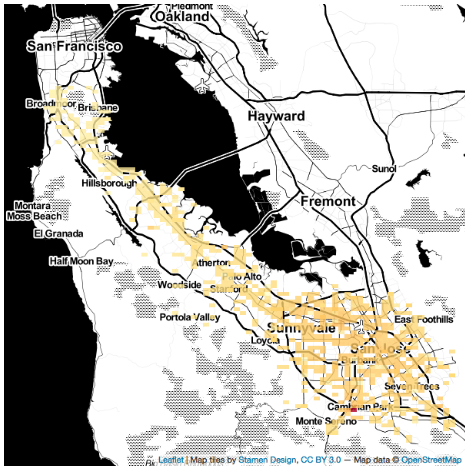
$$\text{Demand}_{i';i} = \sum_u \sum_t P(y_{uti';i} = 1)$$

In Section IV, we repeat this calculation for each restaurant i that either opens or closes. We draw i' from two distinct sets, I_{same} is 100 restaurants chosen at random from the same category as i and I_{diff} is 100 restaurants chosen at random from restaurants that are not in the same category as i. In Table 1 we compare the predicted demand for the place that opens or closes, $\text{Demand}_{i;i}$, to the mean counterfactual predictions for i' in I_{same} and I_{diff}, i.e.,

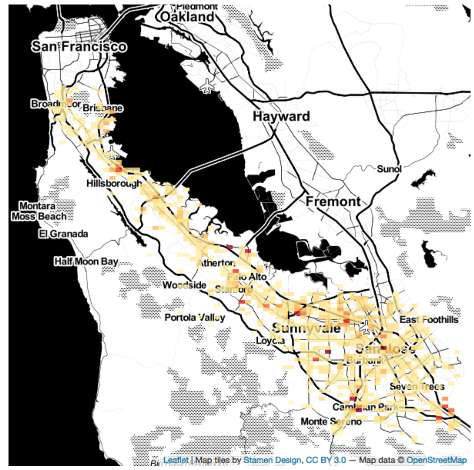
$$\frac{1}{|I_{same}|} \sum_{i' \in I_{same}} \text{Demand}_{i';i}$$

$$\frac{1}{|I_{diff}|} \sum_{i' \in I_{diff}} \text{Demand}_{i';i}$$

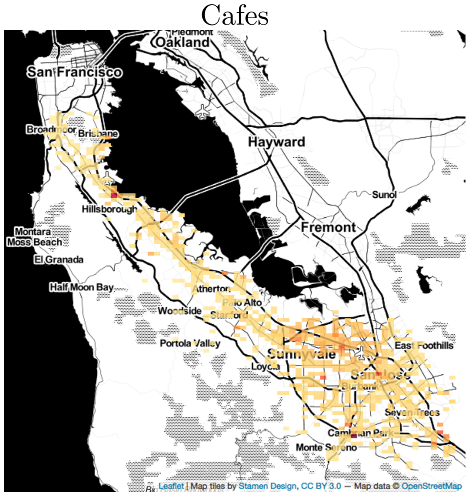
To analyze the best location for a given restaurant category or the best category for a location, the set of target restaurants includes one location selected at random from each geohash6. The set I_{alt} is one restaurant from each major category (the variable `category_most_common`) with the constraint that each restaurant chosen is within 0.1 standard deviation of the population mean for total demand. This constraint was to try to make the set of comparison restaurants relatively similar in popularity. In the "best location for each category" in Figure A13 we plot for a single category i' the predicted demand $\text{Demand}_{i';i}$ for each i in the set of target locations. In Figure A14, we selected subsets of 4 or 5 categories of restaurants from I_{alt} that have the same price range and illustrate for each target location the category of restaurant that is $\text{argmax}_{i' \in I_{alt}} \text{Demand}_{i';i}$.



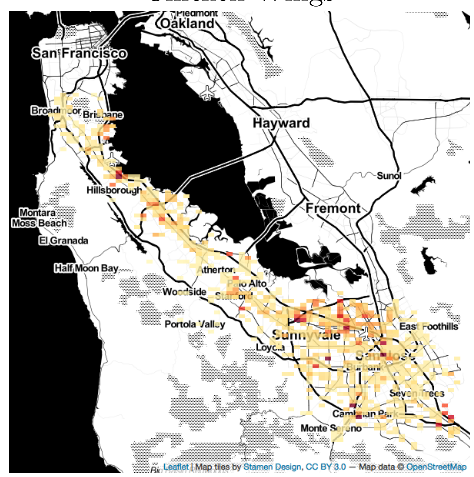
Cafes



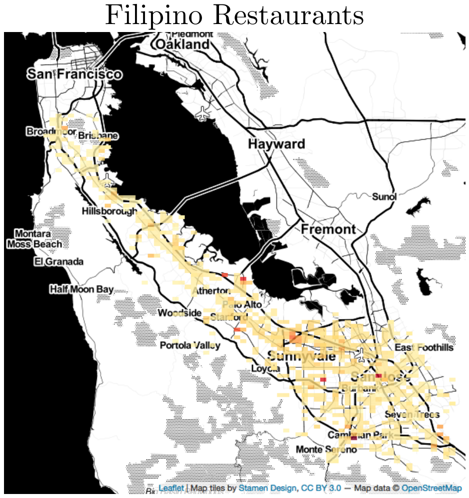
Chicken Wings



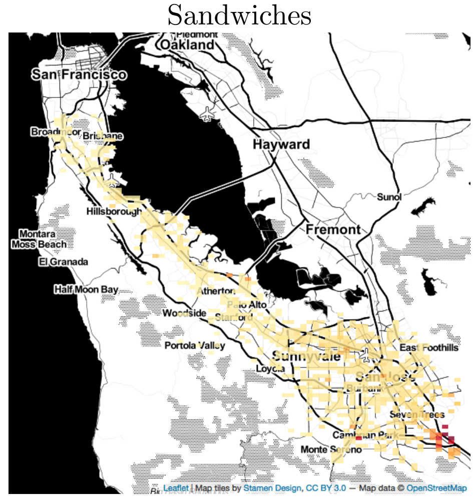
Filipino Restaurants



Sandwiches

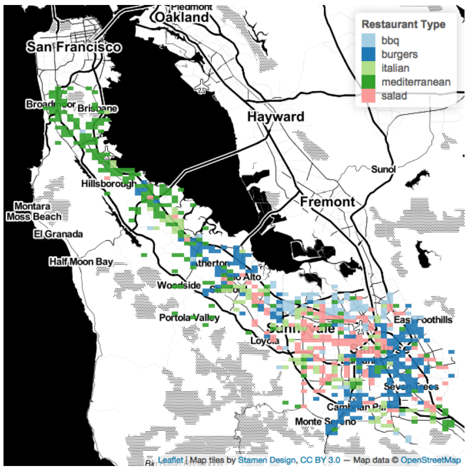


Vegetarian

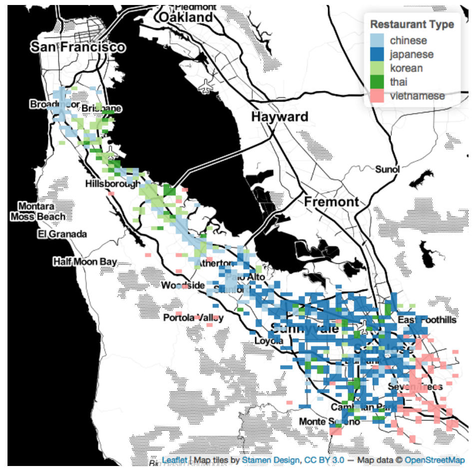


Vietnamese Restaurants

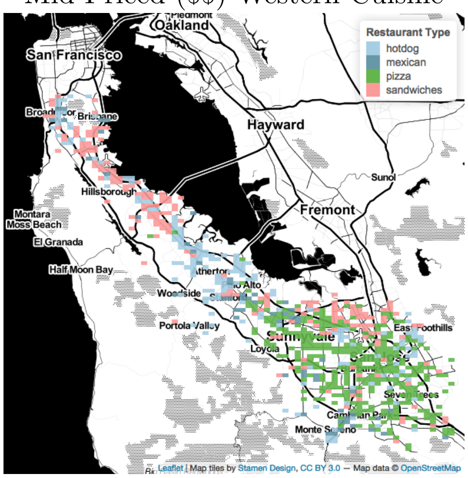
Figure A13. Best Locations for Restaurant Category



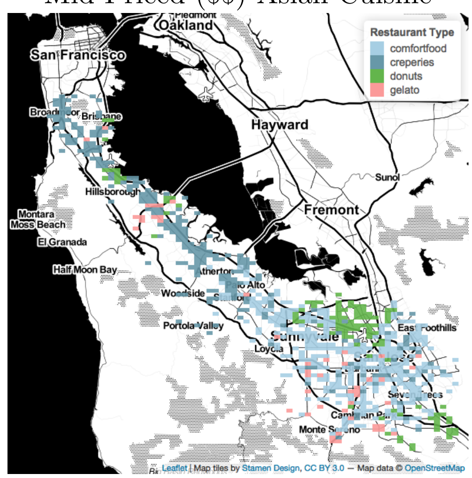
Mid-Priced ($$) Western Cuisine



Mid-Priced ($$) Asian Cuisine



Cheap ($) Fast Food



Cheap ($) Treats

FIGURE A14. BEST RESTAURANT CATEGORY FOR LOCATIONS

*

REFERENCES

Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. 2017. "Variational Inference: A Review for Statisticians." *Journal of the American Statistical Association*, 112(518): 859–877.

Blum, Julius R. 1954. "Approximation methods which converge with probability one." *The Annals of Mathematical Statistics*, 25(2): 382–386.

Bottou, L., F. E. Curtis, and J. Nocedal. 2016. "Optimization Methods for Large-Scale Machine Learning." *arXiv:1606.04838*.

Hoffman, M. D., David M. Blei, C. Wang, and J. Paisley. 2013. "Stochastic Variational Inference." *Journal of Machine Learning Research*, 14: 1303–1347.

Jordan, Michael I., ed. 1999. *Learning in Graphical Models.* Cambridge, MA, USA:The MIT Press.

Kingma, Diederik P., and Max Welling. 2014. "Auto-Encoding Variational Bayes." *arXiv:1312.6114*.

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra. 2014. "Stochastic backpropagation and approximate inference in deep generative models." Vol. 32 of *Proceedings of Machine Learning Research*, 1278–1286. PMLR.

Robbins, H., and S. Monro. 1951. "A stochastic approximation method." *The Annals of Mathematical Statistics*, 22(3): 400–407.

Ruiz, Francisco J. R., Susan Athey, and David M. Blei. 2017. "SHOPPER: A Probabilistic Model of Consumer Choice with Substitutes and Complements." *arXiv:1711.03560*.

Titsias, M. K., and M. Lázaro-Gredilla. 2014. "Doubly stochastic variational Bayes for non-conjugate inference." Vol. 32 of *Proceedings of Machine Learning Research*, 1971–1979. PMLR.

Wainwright, M. J., and M. I. Jordan. 2008. "Graphical Models, Exponential Families, and Variational Inference." *Foundations and Trends in Machine Learning*, 1(1–2): 1–305.

Zhao, He, Lan Du, and Wray Buntime. 2017. "Leveraging Node Attributes for Incomplete Relational Data." Vol. 70 of *Proceedings of Machine Learning Research*, 4072–4081. PMLR.