Biometrics 000, 000–000
000 0000

DOI: 000

# Unifying NYC Taxicab Records and Hotel Occupancy Data

**Daniel J. Saunders**

College of Computer and Information Sciences, University of Massachusetts, Amherst, Massachusetts
*email:* djsaunde@cs.umass.edu

**and**

**Christian Rojas* and Debi Mohapatra****

Department of Resource Economics, University of Massachusetts, Amherst, Massachusetts
**email:* rojas@resecon.umass.edu
***email:* dmohapatra@umass.edu

Summary: A method for fast distributed processing of New York City taxicab trip records with respect to a secondary dataset of daily hotel occupancy and pricing information is presented. We motivate and develop an algorithm used to select an optimal distance threshold for capturing taxi trips relevant to predicting hotel occupancy and pricing rates. Two hotel outlier removal techniques based on distribution matching and simple outlier detection are discussed, and used to reduce prediction error by removing noisy examples from the taxi trips data. The effect of predicting daily hotel occupany rates and pricing with and without conditioning on daily nearby taxi traffic is demonstrated. The strength of this effect after removing hotels with atypical numbers of nearby taxi traffic is also assessed. Further applications of the taxi trip records dataset and other data sources are discussed as sources of potential future work.

Key words: Applied economics; Data analysis; Distributed computing.

## 1. Introduction

The availability of various forms of empirical commercial and industrial data is crucial for the solution of real-world economics problems. At the scale of cities and nations, however, certain data present major data processing challenges. With the advent of massively parallel and relatively cheap distributed computing, however, methods for the processing and analysis of such datasets are not entirely out of reach.

The New York City (NYC) Taxi & Limousine Commission (TLC) Trip Record Data consists in part of yellow and green taxicab trip records spanning the years 2009-2017. Recorded attributes include pick-up and drop-off date and time of day, pick-up and drop-off geospatial or zone-coded location, trip distances, fares, and others. Careful exploration and analysis of such a massive observational dataset may result in powerful insights in transportation issues in NYC. Methods developed on this particular dataset may be generalized to other urban settings in which such data is or becomes available.

Several interesting studies have recently been carried out on the NYC taxi trips dataset. However, combining daily city-wide taxi trajectories with related NYC datasets should allow researchers to propose and test more powerful economic hypotheses. We study a unification of the taxi records with the occupancy rates of 178 NYC hotels from the years 2013 to 2016. We are interested in trips which *begin* or *end* within distance $d$ of at least one hotel in the dataset. To find the best choice of distance threshold $d$ over all hotels, we develop an optimization procedure motivated from a simple distribution matching approach. Hotels with abnormal amounts of nearby taxi traffic may be eliminated from consideration during or after the optimization process.

We hypothesize that both hotel occupancy rates and pricing can be more accurately predicted from the density of daily nearby taxicab pick-up and drop-off rides combined with information from the hotel occupancy data than from the hotel occupancy dataset alone. This effect should be strengthened as hotels with atypical nearby taxi trip density are eliminated from consideration. Assuming these is true, we argue that many other hotel metrics may be better estimated conditioned on observed per-day nearby taxi trips. Estimated values may be used as economic evidence for urban policy-making in the hotel and taxi industries, and others besides, depending on the availability of relevant and timely data.

## 2. Related Work

An analytics model is proposed in Ferreira et al. (2013) which allows users to visually query taxi trips. Standard queries about the data are supported, as well as spatio-temporal "origin-destination" queries, allowing users to discover mobility patterns throughout the city. NYC hotspots are identified in Stoyanovich et al. (2017), in which taxi trips are represented as straight-line trajectories from pick-up to drop-off coordinates. Hotspots are used to identify lack of convenient public transportation options, and to suggest the addition of bus routes and ride-sharing options. A custom software approach is developed on top of the distributed

processing system Apache Spark, allowing efficient analysis of taxi trajectory graphs as they evolve. In another work, a visual analytics method is proposed to study urban mobility patterns using graph modeling of taxicab trajectories (Huang et al., 2016). The dynamics of Shenzhen taxis are assessed using graph analysis techniques, and are studied at multiple scales using a graph partitioning algorithm to produce regional visual analytics. Users of the system can interactively explore street-level city traffic patterns.

A technique for event-guided exploration of large spatio-temporal data is introduced in Doraiswamy et al. (2014), on the basis that manual exploration of such data is time-consuming and ineffective. Computational topology is used to discover interesting events in the data, and an algorithm is developed to group and index events which can be interactively explored or queried by users. This approach is validated on NYC taxi trips as well as subway service records. An analysis of the NYC taxi trips database in combination with various relevant urban data is presented in Wu et al. (2016). Correlations between taxi and the external datasets are used to the predict traffic dynamics; namely, it it shown that points-of-interest (POIs) can predict regular traffic patterns, geo-tagged tweets can explain traffic caused by atypical events, and weather data may explain abnormal drops in traffic density. Chu et al. (2014) develop a method to discover and analyze information contained in a massive dataset of taxi trajectories. Geospatial coordinates are replaced with traversed street names, which are subsequently modeled with topic modeling tools. The "hidden themes", or the discovered topics from the employed topic models, are used to analyze mobility patterns with a visual analytics system.

The efficient processing of spatio-temporal datasets (e.g, NYC taxi trip records) often requires specialized software and hardware. A novel indexing scheme over spatio-temporal data is developed for use with general-purpose graphics processing units (GPUs) in Doraiswamy et al. (2016). The index allows sub-second query speeds over large spatio-temporal datasets such as the NYC taxi data, and a massive number of tweets collected from Twitter.

## 3. Methods

### 3.1 *Data processing*

The dataset of hotel occupancy rates contains information from January 2013 until present day. The dataset of NYC hotel taxi records contains data from January 2009 until present day; however, geospatial coordinates of pick-up and drop-off locations are only recorded from January 2009 until June 2016. Given these constraints, our experiments consider only those records which lie in the date range of January 2013 until June 2016, approximately 3.5 years of joint data. Our analyses are concerned only with the yellow and green taxi services in NYC, and we discard all data concerning "for-hire vehicle" services (e.g., Uber, Lyft, and others).

We use a supercomputing cluster in which users are provisioned with a maximum of 40 CPU compute nodes. On each node, a Python program is dispatched to *pre-process* a single month's worth of taxicab trip data with respect to the set of NYC hotels of interest. In particular, upon specifying a *distance criterion* $d$, each process will independently extract trips from its designated month of data which begin or end with $d$ feet of any of the hotels under consideration. Geospatial distance calculations are accomplished using Vincenty's formula (Bessel et al., 2010). Fast loading and processing of the taxi trip records is accomplished using the `dask` parallel computation and task scheduling Python library (Arabie and Carroll, 2016).

Prior to pre-processing, each month of yellow and green taxi trip data requires approximately 2.5Gb and 300Mb of disk space, respectively, totalling ≈120Gb over the course of the experimental date range. Pre-processing the dataset using a distance criterion $d = 300$ feet reduces this to two files of size ≈10Gb each, one of trips beginning near hotels of interest (*nearby pick-ups*), and another ending nearby (*nearby drop-offs*). This represents an approximate reduction of 83% of the original data.

Pre-processing a month of yellow taxi data using the aforementioned Python program and distance criterion $d = 300$ requires *at most* 40 minutes and 50Gb of random-access memory. Using the super-computing cluster with a 40-node per-user allotment, processing all 42 months of taxi trip date requires only ≈80 minutes. If equipped with 42 or more nodes with the same computing resources, the processing time would be cut in half.

### 3.2 *Distance criterion optimization*

A crucial assumption in the analyses of the taxi dataset is that taxi trips that originate or culminate near a hotel are likely to indicate that guests are arriving or leaving the hotel. With this in mind, we can use the per-hotel densities of nearby taxi trip pick-ups or drop-offs to make predictions about hotel room demand, pricing, or other variables of interest. We additionally assume that some unknown distance threshold $d$ maximizes the likelihood, over all hotels, that the nearby taxi traffic does indeed indicate they are hotel guests. If $d$ is too small, too few guests are captured in the trip coordinates data; if $d$ is too large, too many taxi customers are mis-classified as hotel guests.

We suggest that the distance criterion $d$ should be selected such that it maximizes the predictability of some quantity of interest. We compare a dataset of hotel occupancy information with counts of nearby taxi trips, per hotel and per day. We denote the observed per-hotel distribution of hotel occupancy as $\hat{p}_{\text{occ}}$ and the observed per-hotel distribution of nearby taxi trips as $\hat{p}_{\text{taxi}}$. In our experiments, these distributions are estimated using per-hotel proportions of aggregated hotel occupancy and taxi data from January 1st, 2013 until June 30th, 2016, and the distribution of nearby taxi trips depends on the choice of distance criterion $d$.

We want to choose $d$ such that some divergence measure $\mathcal{D}$ between empirically observed distributions $\hat{p}_{\text{occ}}$ and $\hat{p}_{\text{taxi}}$ is minimized. We consider relative entropy,

$$\sum_x \hat{p}_{\text{occ}}(x) \log \frac{\hat{p}_{\text{occ}}(x)}{\hat{p}_{\text{taxi}}(x)},$$

summed absolute differences,

$$\sum_x |\hat{p}_{\text{occ}}(x) - \hat{p}_{\text{taxi}}(x)|,$$

and summed relative differences, in which the smaller-valued proportion is used as the denominator in the quotient,

$$\sum_x \frac{\min\{\hat{p}_{\text{taxi}}(x),\ \hat{p}_{\text{occ}}(x)\}}{\max\{\hat{p}_{\text{taxi}}(x),\ \hat{p}_{\text{occ}}(x)\}}.$$

Note that the divergence measures used are not necessarily true $f$-divergences. These measures are considered useful on the basis of their predictive power in modeling hotel room demand and pricing.

After pre-processing the NYC taxi data using a suitably large distance criterion (e.g., $d = 300$ft.), a range of candidate distance criterions are considered (e.g., $d \in [25\text{ft.}, 50\text{ft.}, ..., 275\text{ft.}, 300\text{ft.}]$), whose corresponding datasets are subsets of the pre-processed 300ft. dataset. The empirical per-hotel distribution of nearby pick-up and / or drop-off taxi trips $\hat{p}_{\text{taxi}}$ is computed from the data for each choice of $d$, and we calculate the selected divergence measure with respect to the fixed $\hat{p}_{\text{occ}}$; that is, the observed per-hotel distribution of rented rooms from the same time period. The criterion $d$ giving the minimal divergence value is said to be the "best", and the dataset obtained from this choice of $d$ can be used for downstream analyses.

### 3.3 *Removing outlier hotels*

Certain hotels under consideration may have an abnormally large or small number of nearby taxicab pick-ups or drop-offs depending on their location in the city. For example, Hotel Pennsylvania, located adjacent to the Pennsylvania railroad station, has a disproportionately large number of both pick-up and drop-off taxi trips due in part to the traffic that the train station causes. This observation makes it difficult to justify using such hotels in our analyses, as they may confound conclusions about hotels with atypical shares of nearby taxi traffic.

Using the same divergence measures as listed above, we propose two methods for the removal of outlier hotels:

1. Iteratively remove hotels from the occupancy dataset during the distance optimization process until the divergence measure between distributions, $\mathcal{D}(\hat{p}_{\text{occ}}||\hat{p}_{\text{taxi}})$, is below some threshold. Hotels are removed on the basis of how poorly their nearby taxi trip proportion matches their proportion of rented rooms. Picking the threshold is a hyper-parameter choice, and may be difficult to select *a priori*. We may also consider removing a fixed number of hotels based on the same distribution matching procedure. This approach is assessed in Section 4.1.
2. Using a trained machine learning model, hotels may be removed from consideration on the basis of which cause the most prediction error. We may iterate this process until $R^2$ scores have stabilized, or until the model has reached a satisfactory mean-squared error value.

### 3.4 *Predicting hotel occupancy and pricing*

We establish a simple estimation baseline using only information from the hotel occupancy dataset to predict daily hotel room demand. An ordinary least squares (OLS) regression model is fit to the hotel occupany data, where it is assumed that room demand and pricing is a linear function of the hotel identity, the day of the week, the month, and the year. This information allows the model to discriminate typical per-hotel demand, as well as to capture temporal trends at multiple timescales. A multi-layer perceptron (MLP) regression model is also fit to this data, with hyper-parameters (hidden layer sizes and regularization constant) selected by grid hyper-parameter search and averaged three-fold cross-validation loss. Importantly, this model is able to learn features codifying non-linear interactions between the observational data.

To evaluate the effect of conditioning hotel occupany predictions on available nearby taxi trip densities, we include both nearby pick-up and drop-off trips as separate features in both the OLS and MLP models, as described above. Again, hyper-parameters are selected according to a grid search and averaged three-fold cross-validation loss, and the dataset is partitioned into train and test subsets as before. We train and test models using a range of distance thresholds $d \in \{25\text{ft.}, 50\text{ft.}..., 300\text{ft.}\}$, in order to validate the distance thresholds $d$ predicted by the optimization method and chosen divergence measures described above.

All data samples are randomly permuted to avoid leaving any hotels out of any data partitions, and 5 independent realizations of the chosen models are trained and tested. Averaged $R^2$ and mean-squared error (MSE) of both models are reported for a variety of settings of distance criterions $d$ and with and without outlier hotels removed in Section 4.2.

All machine learning models are fit and evaluated using the `scikit-learn` machine learning library (Pedregosa et al., 2011).

## 4. Results

### 4.1 *Outlier removal: choosing optimal distance threshold*

Using the first outlier removal method, as part of the distance criterion optimization procedure, we may create tables of the order of hotel removals from the dataset for each divergence measure. We include one such table, in which the absolute differences measure is used to decide the hotel removal order. Included also are the removed hotels' absolute difference values, as well as the optimal distance criterion $d$ for that particular iteration of the removal algorithm. Though there are 178 hotels in the dataset, we truncate the table after just 15 iterations in order to give an idea of the removal behavior using the relative difference measure.

Throughout the rest of the optimization, the optimal distances $d$ produced on each iteration typically fell in the range $[175, 250]$. We may check whether these choice of distances produces subsets of taxi trip data which, when used as features in a machine learning algorithm, give better hotel occupancy and pricing prediction accuracy. The same may be investigated for the range of distances given by the other considered divergence measures. In this context, the most useful divergence measure will yield subsets of taxi data which, when

Table 1: Order of hotel removals and corresponding data using the relative difference divergence measure.

| Removal | Hotel | $\hat{p}_{\text{occ}}$ | $\hat{p}_{\text{taxi}}$ | Best $d$ | Rel. diff. |
|---|---|---|---|---|---|
| 1 | Res. Inn ... Trade Center | $6 \times 10^{-7}$ | $2.6 \times 10^{-3}$ | 300 | 4,549 |
| 2 | Hotel Gansevoort | $3 \times 10^{-3}$ | $2 \times 10^{-2}$ | 190 | 6.33 |
| 3 | Courtyard ... Herald Sqr. | $2.2 \times 10^{-3}$ | $1.3 \times 10^{-2}$ | 180 | 5.93 |
| 4 | Hilton ... Park Avenue | $5.7 \times 10^{-4}$ | $3.3 \times 10^{-3}$ | 180 | 5.74 |
| 5 | Hotel On Rivington | $1.7 \times 10^{-3}$ | $9.6 \times 10^{-3}$ | 195 | 5.62 |
| 6 | Holiday Inn Express ... | $4.3 \times 10^{-3}$ | $8 \times 10^{-4}$ | 195 | 5.28 |
| 7 | Hilton NY Midtown | $3.6 \times 10^{-2}$ | $7.4 \times 10^{-3}$ | 195 | 4.91 |
| 8 | Doubletree ... Fin. Distr. | $7 \times 10^{-3}$ | $1.6 \times 10^{-3}$ | 190 | 4.27 |
| 9 | Sohotel | $9 \times 10^{-4}$ | $3.8 \times 10^{-3}$ | 190 | 4.2 |
| 10 | Res. Inn ... Central Park | $2 \times 10^{-3}$ | $8.1 \times 10^{-3}$ | 180 | 4.02 |
| 11 | Hilton ... Square Central | $9.2 \times 10^{-4}$ | $3.6 \times 10^{-3}$ | 180 | 3.87 |
| 12 | Marriott NY Marquis | $3.8 \times 10^{-2}$ | $9.9 \times 10^{-3}$ | 180 | 3.8 |
| 13 | Sheraton ... Times Square | $3.5 \times 10^{-2}$ | $9.5 \times 10^{-3}$ | 190 | 3.72 |
| 14 | Fairfield ... Penn Station | $4.2 \times 10^{-3}$ | $1.1 \times 10^{-3}$ | 195 | 3.65 |
| 15 | Holiday Inn ... 57th St. | $1.2 \times 10^{-2}$ | $3.5 \times 10^{-3}$ | 195 | 3.5 |

Table 2: MSE and $R^2$ values for OLS and MLP regression models trained without relevant taxi data.

| Model | MSE (train) | $R^2$ (train) | MSE (test) | $R^2$ (test) |
|---|---|---|---|---|
| OLS | 106,552 ± 481 | 0.0166 | 105,463 ± 1,923 | 0.0165 |
| MLP | 79,592 ± 3,382 | 0.2620 | 80718 ± 5113 | 0.2610 |

used as observations in a machine learning model, reliably produce the most accurate predictions.

### 4.2 *Predicting hotel occupancy and pricing*

We first establish a baseline hotel occupancy and pricing learning setup using only the day of the week, the date, and the identity of the hotel. Given these observations, a model is trained to output the daily hotel occupancy and room pricing, both integer targets. The data is randomly shuffled to avoid leaving hotels out of any particular subset, and split into 80%, 20% training, test partitions. An OLS model is fit to the training data and evaluated on the test data. A multi-layer perceptron regression model is also fit to the training data, where network hyper-parameters are chosen via random search according to the best predictive accuracy on the validation data, and is evaluated on the test data.

The results for both models are given in Table 2.

4.2.1 *Full hotel data.* To compare, the same regression models are fit to the same observations, albeit with the counts of nearby pick-up and / or drop-off taxi trips per hotel and per day. In this first modeling attempt, we utilize the full dataset of hotels, regardless of how atypical their nearby pick-up and / or drop-off taxi trip counts may be.

We train and evaluate all models using nearby taxi trip data using distances $d \in [25\text{ft.}, 50\text{ft.}, \ldots, 275\text{ft.}, 300\text{ft.}]$, in order to test predictions from the distance optimization. In all experiments, we report the values of both (training and test) mean squared error (MSE) and the coefficient of determination, $R^2$.

The results for the OLS regression model and all considered settings of $d$ are given in Table 3. Although OLS models fit along with taxi data demonstrate a better fit in terms of $R^2$ values, there is no improvement in mean-squared error, and even a significant increase when $d$ is sufficiently small.

Table 3: MSE and $R^2$ values for OLS regression model trained with relevant taxi data with a range of distance thresholds $d$.

| Model | $d$ (ft.) | MSE (train) | $R^2$ (train) | MSE (test) | $R^2$ (test) |
|---|---|---|---|---|---|
| OLS | 25 | 117,339 ± 499 | 0.0193 | 116,856 ± 1,990 | 0.0193 |
| | 50 | 108,431 ± 631 | 0.0160 | 105,588 ± 2,524 | 0.0154 |
| | 75 | 106,501 ± 258 | 0.0167 | 106,314 ± 1,030 | 0.0167 |
| | 100 | 108,146 ± 454 | 0.0211 | 106,199 ± 1,818 | 0.0211 |
| | 125 | 107,166 ± 549 | 0.0215 | 107,298 ± 2,196 | 0.0217 |
| | 150 | 107,491 ± 693 | 0.0213 | 105,996 ± 2,779 | 0.0223 |
| | 175 | 107,277 ± 684 | 0.0212 | 106,929 ± 2,740 | 0.0228 |
| | 200 | 107,378 ± 495 | 0.0214 | 106,525 ± 1,979 | 0.0220 |
| | 225 | 107,256 ± 525 | 0.0212 | 106,999 ± 2,098 | 0.0226 |
| | 250 | 107,251 ± 411 | 0.0213 | 107,006 ± 1,643 | 0.0223 |
| | 275 | 106,844 ± 506 | 0.0218 | 108,628 ± 2,023 | 0.0205 |
| | 300 | 107,212 ± 634 | 0.0214 | 107,157 ± 2,538 | 0.0220 |
| MLP | 25 | 60,384 ± 8,877 | 0.4994 | 60,428 ± 9,796 | 0.4967 |
| | 50 | 50,176 ± 5,267 | 0.5476 | 50,122 ± 5,589 | 0.5530 |
| | 75 | 43,835 ± 4,942 | 0.6014 | 44,454 ± 4,662 | 0.5985 |
| | 100 | 45,588 ± 5,509 | 0.5851 | 45,779 ± 5,020 | 0.5872 |
| | 125 | 47,437 ± 4,877 | 0.5673 | 47,630 ± 5,204 | 0.5637 |
| | 150 | 50,678 ± 10,050 | 0.5379 | 50,648 ± 9,794 | 0.5355 |
| | 175 | 46,218 ± 6,021 | 0.5788 | 46,501 ± 6,034 | 0.5730 |
| | 200 | 50,535 ± 12,527 | 0.5372 | 51,004 ± 13,138 | 0.5402 |
| | 225 | 37,277 ± 6,933 | 0.6603 | 37,064 ± 6,297 | 0.6593 |
| | 250 | 47,705 ± 13,422 | 0.5650 | 47,548 ± 11,538 | 0.5642 |
| | 275 | 35,077 ± 1,456 | 0.6809 | 35,401 ± 1,668 | 0.6727 |
| | 300 | 38,173 ± 5,142 | 0.6514 | 38,251 ± 4,996 | 0.6513 |

The results for the MLP regression models with the considered range of distances $d$ are also given in Table 3. Even with the smallest considered subset of taxi data given by choosing the distance threshold $d = 25$ft. produces a significant decrease in training and test mean-squared error, and a significantly higher value of $R^2$. Though there is some variability in the estimator goodness of fit as $d$ is increased, MSE values tend to decrease while $R^2$ values increase. With all hotels included, best candidate distance criterions include $d = 225$ft., 275ft., and 300ft.

4.2.2 *With hotel outlier removal.* Using recommendations from both outlier detection approaches, we investigate whether removing certain hotels from the dataset produces qualitatively better predictions, as measured by test data $R^2$ values.

## 5. Conclusion


### Acknowledgements

The authors would like to thank the College of Computer and Information Sciences for their gracious allowance of supercomputing resources.


### References


Arabie, P. and Carroll, J. D. (2016). *Dask: Library for dynamic task scheduling.*

Bessel, F. W., Karney, C. F. F., and Deakin, R. E. (2010). The calculation of longitude and latitude from geodesic measurements. *Astronomische Nachrichten* **331,** 852–861. arXiv: 0908.1824.

Chu, D., Sheets, D. A., Zhao, Y., Wu, Y., Yang, J., Zheng, M., and Chen, G. Z. (2014). Visualizing hidden themes of taxi movement with semantic transformation. *2014 IEEE Pacific Visualization Symposium* pages 137–144.

Doraiswamy, H., Ferreira, N., Damoulas, T., Freire, J., and Silva, C. T. (2014). Using topological analysis to support event-guided exploration in urban data. *IEEE Transactions on Visualization and Computer Graphics* **20,** 2634–2643.

Doraiswamy, H., Vo, H. T., Silva, C. T., and Freire, J. (2016). A gpu-based index to support interactive spatio-temporal queries over historical data. *2016 IEEE 32nd International Conference on Data Engineering (ICDE)* pages 1086–1097.

Ferreira, N., Poco, J., Vo, H. T., Freire, J., and Silva, C. T. (2013). Visual exploration of big spatio-temporal urban data: A study of new york city taxi trips. *IEEE Transactions on Visualization and Computer Graphics* **19,** 2149–2158.

Huang, X., Zhao, Y., Ma, C., Yang, J., Ye, X., and Zhang, C. (2016). Trajgraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data. *IEEE Transactions on Visualization and Computer Graphics* **22,** 160–169.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12,** 2825–2830.

Stoyanovich, J., Gilbride, M., and Moffitt, V. Z. (2017). Zooming in on nyc taxi data with portal. *CoRR* **abs/1709.06176,**.

Wu, F., Wang, H., and Li, Z. (2016). Interpreting traffic dynamics using ubiquitous urban data. In *SIGSPATIAL/GIS*.