BiasBuster: a Neural Approach for Accurate Estimation of Population Statistics using Biased Location Data

Sepanta Zeighami*
UC Berkeley
zeighami@berkeley.edu

Cyrus Shahabi
University of Southern California
shahabi@usc.edu

***Abstract*—While extremely useful (e.g., for COVID-19 forecasting and policy-making, urban mobility analysis and marketing, and obtaining business insights), location data collected from mobile devices often contain data from a biased population subset, with some communities over or underrepresented in the collected datasets. As a result, aggregate statistics calculated from such datasets (as is done by various companies including Safegraph, Google, and Facebook), while ignoring the bias, leads to an inaccurate representation of population statistics. Such statistics will not only be generally inaccurate, but the error will disproportionately impact different population subgroups (e.g., because they ignore the underrepresented communities). This has dire consequences, as these datasets are used for sensitive decision-making such as COVID-19 policymaking. This paper tackles the problem of providing accurate population statistics using such biased datasets. We show that statistical debiasing, although in some cases useful, often fails to improve accuracy. We then propose BiasBuster, a neural network approach that utilizes the correlations between population statistics and location characteristics to provide accurate estimates of population statistics. Extensive experiments on real-world data show that BiasBuster improves accuracy by up to 2 times in general and up to 3 times for underrepresented populations.**

***Index Terms*—Location bias, estimation, machine learning**

I. Introduction

Location data collected from mobile devices is useful: [1]–[4] use such datasets for COVID-19 forecasting and policy-making, [5]–[7] use them for urban mobility analysis and [8]–[10] use them for marketing and obtaining business insights. A concrete example is deciding closure policies for COVID, which depends on how many people go to a location and how long they stay there (a place should be closed if many people stay there for a sufficiently long duration). Such studies use aggregate statistics obtained from the observed data. The observed data is a set of location signals (e.g., GPS pings from cellphones) from which aggregate statistics are calculated (e.g., number of people in an area, average time people stay at a location and average distance they travel to get there).

However, observed location data is often a biased subset of the population data. More data is available for some sub-groups of the population, a bias that occurs due to the means of data collection. Location datasets are collected from mobile apps, which are often used by different sub-groups within a population at varying degrees. Android phones and iPhones are used by different demographics [11], [12], and dependence on demographics such as age is more broadly true across different app users [13], [14]. To quantify such biases, we analyzed the location dataset used by Safegraph, a popular data curator, which provides aggregate location-based statistics [15] extensively used for COVID studies [1]–[4] and other applications [5]–[10]We observe that the dataset contains more data for the adult and low-income populations (we suspect this is due to the data being mostly collected from Android phones). while it contains less data for the senior or child populations. This is summarized in Table I, which shows the Pearson correlation between the portion of the neighbourhood's population for which data was available and various neighbourhood demographics (see Sec. V for methodology and details). A higher correlation for an attribute (e.g., income) means neighbourhoods with a higher value of that attribute had a larger portion of their population in the observed data. For instance, in Fargo, we see a relatively strong negative correlation between income and the population's representation in the observed GPS data.

City	Adult	Senior	Child	Median Income
Houston	0.31	-0.12	-0.24	-0.08
Chicago	0.09	-0.02	-0.19	-0.18
San Francisco	0.16	0.04	-0.30	-0.24
Tulsa	0.41	-0.12	-0.22	-0.29
Fargo	0.63	-0.38	-0.40	-0.39

Table I: Correlation between observed proportion of neighbourhood population and neighbourhood demographics

The problem, then, is to provide accurate aggregate population statistics while having access only to a biased sample of the population's location. An approach that reports aggregate statistics but is oblivious to the present bias leads to inaccurate estimates. In this case, the estimation error is due to using a *biased estimator* of the population statistic. Consequently, we not only obtain inaccurate estimations in general, but the estimation error will also disproportionately impact different population subgroups. For instance, less data for the senior population leads to larger errors for such a population, while more data for people with low income can lead to an overestimation of densities in low-income neighbourhoods (because

*Work done while the author was a PhD student at USC's Infolab

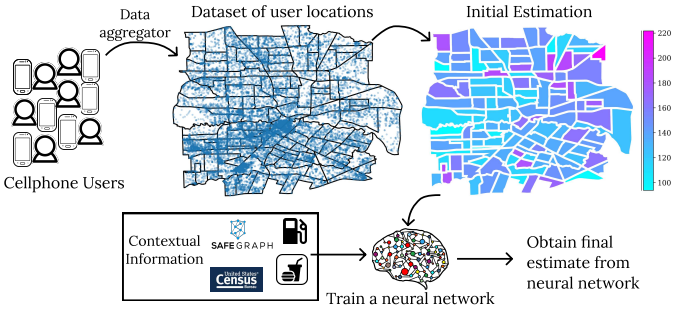

Figure 1: BiasBuster end-to-end pipeline

low-income neighbourhoods will appear to be more densely populated compared with other neighoubhoods in our observed data). This can, for example, result in COVID-19 policies that put undue burdens on such (often more vulnerable) populations. It is, therefore, essential to provide accurate estimates for all population subgroups.

To improve the accuracy, one approach is to use statistical debiasing to provide an *unbiased estimator* of the population statistics (i.e., an estimator equal to the population statistic on expectation). This is achieved by utilizing the probabilities of users being sampled, and weighting the samples according to those probabilities when estimating the population statistics. Although such an approach eliminates the bias in estimation, the variance of the estimator (due to the randomness in sampling) leads to errors in the estimations. We observed that debiasing helped improve the accuracy when calculating `COUNT` statistics (e.g., number of users at a location), but led to worse estimates for `AVG` statistics (e.g., average duration people stay at a location). Indeed, in our datasets, we observed the variance of attributes such as visit duration and distance traveled were much larger than the variance of the number of visits. For example, many users have a similar number of visits to a location (e.g., most people go to the same restaurant at most once a week), while there are variations across how long people stay there (e.g., short stays to pick-up food, longer stay to eat and even longer stays for social gatherings). This leads to larger variance when estimating `AVG` queries, which is further amplified by debiasing as often samples needed to be weighted by large values. As a result, we observe an example of the bias/variance trade-off, where debiasing eliminates the bias but the increased variance of the estimator leads to a larger error for `AVG` statistics.

In this paper, we introduce BiasBuster, a learned estimator that utilizes patterns in the aggregate statistics to provide accurate estimates of population statistics for all subgroups of the population. The end-to-end pipeline is presented in Fig. 1. Given a set of location data collected from mobile users, BiasBuster first uses statistical methods described above to obtain initial estimates for population statistics. Then, it uses these initial estimates, together with contextual information available about different locations to train a model that learns the underlying correlations between location characteristics and the population statistics (e.g., people tend to stay shorter in gas stations than in restaurants). Learning such correlations leads to a model that provides accurate estimation for the population statistics. We experimentally verify this observation, showing that BiasBuster reduces the estimation error by a factor of 2 in general, and specifically reduces the error by a factor of 3 for under-sampled neighbourhoods.

BiasBuster uses the inductive bias that correlations between location characteristics and population statistics exist, and a learned model is used to capture such correlations to reduce the error. For instance, people tend to spend a similar amount of time in similar locations (e.g., gas stations), and the model will be able to aggregate such information across locations to reduce the error across locations. Furthermore, the correlations between location characteristics and population statistics can be learned from locations where accurate estimates are available and extrapolated to locations for which less data is available. This takes advantage of the fact that more data is available for some neighbourhoods, and allows improving the accuracy for neighbourhoods where lack of data would otherwise have led to large errors, thus providing accurate answers across the board.

Specifically, our contributions are as follows.

- We present the first study of location bias and its impact on releasing aggregate statistics using large-scale GPS datasets.
- We present, BiasBuster, a neural network approach that provides accurate estimates of population statistics by taking into account contextual information
- Through extensive experiments on commonly used real-world datasets, we show BiasBuster improves accuracy by up to 3 times for under-represented neighbourhoods and 2 times across all neighbourhoods, while, surprisingly, statistical debiasing often worsens the accuracy.

The rest of this paper is organized as follows. Sec II discusses related work, Sec. III describes problem setting, Sec. IV discusses our methodology, Sec. V presents our empirical study and Sec. VI concludes the paper.

II. Related Work

Bias in observed location data has been documented in a variety of data sources [16]–[20], [20], [21]. Closest to our setting, [19] reports similar biases on Safegraph data, showing that older and non-white populations are less likely to be captured by mobility data. Their analysis compares aggregate statistics released by Safegraph to voter turnout records to uncover biases. Although different in methodology and data sources (they use voter turnout records as population ground truth while we utilize census data), their results corroborate our observations regarding the existing biases in the dataset. Nonetheless, we note that Safegraph does provide an analysis of bias in their datasets [22], where they argue that on aggregate and across the US, the proportion of the observed population in certain demographics across the US is the same as the proportion of the true population in that demographic across the US. As also noted by [19], this observation does not imply that there is no sampling bias, because biases at lower granularity can still exist. This is shown in our experiments,

where we observe that sampling ratio of census tracts is negatively correlated with their median income. We also observe that sampling ratios are correlated with age groups (similar observation is reported by [19]), a demographic attribute not considered by Safegraph [22] in their analysis. Overall, we have focused on Safegraph dataset because we have access to the raw data (unaggregated GPS signals, see Sec. V) used for calculating aggregate statistics, but we expect bias in location data to be pervasive across the board and in any dataset collected through mobile apps.

To the best of our knowledge, there is no existing work that addresses bias in location data, with the related work being confined to only documenting such biases. Although many companies have released such aggregate statistics, (e.g., Google Mobility Report [23], Facebook Social Connectedness Index [24] in addition to Safegraph Patterns [15]), the statistics are reported by simply aggregating the observed data without accounting for the bias. Related to our setting is the work on synthetic trajectory generation such as [25], [26], which can be used to generate more data to increase the size of the observed data. However, such approaches will not be able to account for the observed bias, and will merely replicate such biases while creating more data. Moreover, our use of neural networks for estimating population statistics follows the NeuroDB framework [27], [28] and is similar to [29], [30] and [27] that, respectively, do so to answer queries in a privacy-preserving manner or for incomplete relational databases. Finally, the use of machine learning for estimating population statistics is not new [31], [32], but to the best of our knowledge, this is the first work studying how to address bias in location data using machine learning.

III. Problem Setup

Setup. We are given a dataset of user stay-point sequences, D, of n users, where each user's sequence describes locations the user has stayed at in a city. This stay-point sequence is derived from user trajectories (i.e., raw GPS readings) by extracting the locations at which the users stayed for a long enough duration (we describe the exact methodology in Sec. V). For a user u, their stay-point sequence is a sequence of k_u stay-points, $< p_1, ..., p_{k_u} >$, for an integer k_u. A stay-point, p, is a tuple $p =$(`lat`, `lon`, `arrive_t`, `leave_t`), where `lat` and `lon` denote the latitude and longitude where the user stayed, `arrive_t` is the time the user entered the location and `leave_t` the time the user left the location. We assume D is a subset of a set $\mathcal{D}$, where $\mathcal{D}$ is the set of stay-point sequences of the entire population of size N. That is, $\mathcal{D}$ is the set of stay-point sequences of all the population of a city, whereas D is the set of stay-point sequences of the subset of the population whose location has been collected.

Sampling Bias. We use observed population statistics (from our dataset) and government-released population statistics (from censuses) for the city to study the sampling procedure.

For each user, we consider their *home neighbourhood* to be the neighbourhood where they spend the most time (i.e., we assume users spend the most time at their home, since they stay there overnight and for long periods). This information is obtained from their list of stay-points. We use the users' home neighbourhoods to compare the observed population of each neighbourhood with its true population. Let $h(u)$ be the home neighbourhood of a user u. Then, the observed population, n_μ, of a neighbourhood μ is $n_\mu = |\{u \in D, h(u) = \mu\}|$.

We consider the government-released statistics for the city to be the true population statistics, i.e., calculated from $\mathcal{D}$. In the US, where our datasets are collected (See Sec. V for dataset details), US Census releases such information. We utilize aggregate population statistics (e.g., population, median income) released for different census tracts, where census tracts are small (with about 4,000 inhabitants [33]) subdivisions of counties (similar to zip codes). We consider each census tract as a neighbourhood and use the terms census tract and neighbourhood interchangeably. Let N_μ be the true population of the neighbourhood μ (obtained from Census).

We define the *sampling ratio*, s_μ, of μ as $s_\mu = \frac{n_\mu}{N_\mu}$, which denotes the fraction of the population from the neighbourhood that is sampled. We consider the setting where the sampling ratio, s_μ, is different for different neighbourhoods μ, the setting we call biased sampling. In this case, for different neighbourhoods, a different portion of their population has been sampled. Biased sampling based on other attributes (e.g., income, race, age) often translates to bias based on neighbourhood, since neighbourhoods in the US are often segregated and homogeneous [34], [35]. Biased sampling happens often in practice; Table I presents one such instance where it shows the dataset used by Safegraph (used by [1]–[10]

Problem Definition. Our goal is to provide an accurate estimate of population statistics given access only to a biased subsample, that is to provide estimates of a statistic using D so that the answer is similar to answers from $\mathcal{D}$. Population statistics are aggregate queries over $\mathcal{D}$. We specifically focus on aggregate queries over neighbourhoods, and we use the terms *queries* and *statistics* interchangeably. A query $q = (AGG, \alpha, \mu)$ asks for aggregation, AGG, of an attribute, α, of all the stay-points, p, that fall in neighbourhood μ, so that the true answer to a query q is $AGG(\{p[\alpha] | p \in u_\mu, u \in \mathcal{D}\})$, where for a user u, we denote their set of stay-points in μ by u_μ, i.e., $u_\mu = \{p, (p[\text{lat}], p[\text{lon}]) \in \mu\}$. Although our approach is generic, we consider `COUNT` and `AVG` aggregation functions and specifically queries of *total number of visits*, *average visit duration* and *average distance travelled*. All such statistics are important for disease spread analysis and policy-making, as well as transportation and urban planning, and Safegraph is already releasing aggregate information for these attributes for different neighbourhoods [15]. Specifically, the visit duration of each stay-point p is calculated as p[`leave_t`]−p[`arrive_t`]. Furthermore, for the i-th stay point of a user, p_i, the distance travelled is `dist`((p_i[`lat`], p_i[`lon`]), (p_{i+1}[`lat`], p_{i+1}[`lon`])), where we use Euclidean distance (since distances are short, there is no need to take curvature of the earth into consideration using geographic coordinate system (GCS)) as our distance function (distance travelled for the user's first stay-point is undefined and ignored in computations). We consider both visit duration and distance

travelled to be attributes of the stay-point. The average visit duration and average distance travelled queries ask for the average of visit duration and distance travelled of all the stay-points (across all users) that fall in a neighbourhood, and total number of visits is similarly defined. For a query $q = (AGG, \alpha, \mu)$, we denote by c_μ the true answer to q if $AGG =$COUNT and by y_μ^α if if $AGG =$AVG.

Sampling Assumption and Terminology. Although the following sampling assumptions are not required by our methodology, we use these assumptions to analyze and evaluate different approaches. We assume that the data D is sampled from $\mathcal{D}$ as follows. For every neighbourhood, μ, n_μ i.i.d and uniform samples from $\{u \in \mathcal{D}, h(u) = \mu\}$ are selected, i.e., each sample X_i^μ, $1 \leq i \leq n_\mu$ is equal to one of elements of $\{u \in \mathcal{D}, h(u) = \mu\}$ with probability $\frac{1}{N_\mu}$. This sampling procedure is repeated for all neighbourhoods, where we obtain a different number of samples for different neighbourhoods. In practice, this assumption often holds due to the homogeneity of neighbourhoods [34], [35]. For instance, if an app that collects data is mostly used by the senior population, then more data will be collected from neighbourhoods with a larger older population. However, within the older population, the sampling can be assumed to be uniform.

Therefore, the database D is a collection of random variables. Next, for concreteness, we review some terminology. For an estimator $\hat{\theta}$, calculated from D, to estimate a population statistic θ, recall that bias of $\hat{\theta}$ is $\texttt{bias}(\hat{\theta}) = E[\hat{\theta}] - \theta$. The estimator $\hat{\theta}$ is called an unbiased estimator of θ if $E[\hat{\theta}] = \theta$ and is otherwise called a biased estimator. Furthermore, recall that the mean squared error of an estimator can be written as

$$E[(\theta - \hat{\theta})^2] = \texttt{bias}(\hat{\theta})^2 + \texttt{var}(\hat{\theta}). \quad (1)$$

IV. Querying Biased Location Data

In this section, we present BiasBuster, our neural network approach for answering queries on biased location data. We first discuss the downsides of answering queries oblivious to the bias (Sec. IV-A) or using statistical debiasing (Sec. IV-B). Subsequently, in Sec. IV-C we present our learned approach that addresses such shortcomings.

A. The Oblivious Method

A naive approach to solving the problem is answering queries without considering the bias.

AVG. For AVG queries, this means reporting answers directly on the observed dataset. Specifically, for a query on attribute α in neighbourhood μ the estimate from D is

$$\hat{y}_\mu^\alpha = \frac{\sum_{u \in D} \sum_{p \in u_\mu} p[\alpha]}{\sum_{u \in D} |u_\mu|},$$

Where to calculate the average visit duration for a neighborhood, we go over all the observed visits in the neighborhood and report their average value as the answer.

COUNT. For COUNT queries, we need to scale the query answers observed on the sample dataset. Ignoring the sampling bias, we scale the observed answers by $\frac{N}{n}$ to obtain the estimate

$$\hat{c}_\mu = \frac{N}{n} \sum_{u \in D} |u_\mu|.$$

Shortcomings. Intuitively, in this approach and for COUNT queries, if we observe 10% of the population, we scale our estimates by 10. This is problematic, because, e.g., for a neighbourhood that is mostly visited by seniors, and when the senior population is under-sampled (i.e., less than 10% of the older population is sampled), then scaling by 10 underestimates the number of stay-points for that neighbourhood. A similar example holds for average queries, e.g., if the senior population stays for a shorter duration than the rest of the population in a neighbourhood, then the average calculated based on the observed samples overestimates the true average.

More theoretically, given our sampling assumption it is easy to see that both $\hat{c}_\mu$ and $\hat{y}_\mu^\alpha$ estimators are biased estimators of the true population statistics. Since the error of an estimator can be decomposed into bias and variance terms, this bias can contribute to large errors for such an approach. Note that $\hat{c}_\mu$ and $\hat{y}_\mu^\alpha$ are biased estimators because the sampling procedure is biased and they do not take this bias into account. That is, if the data was sampled uniformly at random from the population, $\hat{c}_\mu$ and $\hat{y}_\mu^\alpha$ would have been unbiased estimators.

B. Statistical Debiasing

To reduce the error, our second attempt uses statistical debiasing to provide unbiased estimators of the population statistics, weighing observed samples with their probability.

COUNT. Consider the users with home neighbourhood η. To answer queries, we weight the visits of each user from η by $\frac{N_\eta}{n_\eta}$. Intuitively, scaling by $\frac{N_\eta}{n_\eta}$ is similar to assuming for every observed user from η there are $\frac{N_\eta}{n_\eta}$ (unobserved) users in η that have the same characteristics. For instance, if the sampling rate for the seniors is 1% while the sampling rate for the rest of the population is 10%, to know how many people are at a location, one needs to scale every observation from seniors by 100 and observations from the rest of the population by 10. Scaling by a larger value helps account for the fact that the older population was under-sampled.

Let H be the set of all neighbourhoods. Formally, our estimate of the number of people in a neighbourhood μ is

$$\hat{c}_\mu = \sum_{\eta \in H} \frac{N_\eta}{n_\eta} \sum_{u \in D, h(u) = \eta} |u_\mu|.$$

Lemma 4.1: $\hat{c}_\mu$ is an unbiased estimator of the population statistic c_μ under the sampling assumptions of Sec. III.

Proof.

$$E[\hat{c}_\mu] = \sum_{\eta \in H} \frac{N_\eta}{n_\eta} \sum_{u \in D, h(u) = \eta} E[|u_\mu|] = \sum_{\eta \in H} \frac{N_\eta}{n_\eta} n_\eta E[|u_\mu|]$$
$$= \sum_{\eta \in H} \frac{N_\eta}{n_\eta} n_\eta \sum_{u_\mu \in \mathcal{D}, h(u) = \eta} \frac{1}{N_\eta} |u_\mu| = c_\mu$$

□

AVG. For average queries, obtaining an unbiased estimator is more difficult. For an attribute α, let the true attribute sum for neighbourhood μ be

$$t_\mu^\alpha = \sum_{\eta \in H} \sum_{u \in \mathcal{D}, h(u) = \eta} \sum_{p \in u_\mu} p[\alpha].$$

The average of the attribute at μ is therefore $y^\alpha_\mu = \frac{t^\alpha_\mu}{c_\mu}$. The difficulty in obtaining an unbiased estimator is due to having to estimate both the numerator and the denominator of this quantity. To simplify the discussion, we assume c_μ is known. To obtain an estimator, we only need to estimate t^α_μ, which can be done by weighting user stay-points similar to COUNT. Specifically, let

$$\hat{t}^\alpha_\mu = \sum_{\eta \in H} \frac{N_\eta}{n_\eta} \sum_{u \in D, h(u)=\eta} \sum_{p \in u_\mu} p[\alpha].$$

Similar to the above, we have that

Lemma 4.2: $\frac{\hat{t}_\mu}{c_\mu} = \hat{y}^\alpha_\mu$ is an unbiased estimator of y^α_μ under the sampling assumptions of Sec. III.

Proof.

$$\begin{aligned} E[\hat{t}_\mu] &= \sum_{\eta \in H} \frac{N_\eta}{n_\eta} \sum_{u \in D, h(u)=\eta} E[\sum_{p \in u_\mu} p[\alpha]] \\ &= \sum_{\eta \in H} \frac{N_\eta}{n_\eta} n_\eta E[\sum_{p \in u_\mu} p[\alpha]] \\ &= \sum_{\eta \in H} \frac{N_\eta}{n_\eta} n_\eta \sum_{u_\mu \in \mathcal{D}, h(u)=\eta} \frac{1}{N_\eta} \sum_{p \in u_\mu} p[\alpha] \\ &= t_\mu. \end{aligned}$$

Therefore, $E[\frac{\hat{t}^\alpha_\mu}{c_\mu}] = y^\alpha_\mu$. □

In practice, we observed that even with the assumption that c_μ is known, this unbiased estimator performs poorly, so we do not further relax this assumption. Nonetheless, we note that the estimator $\frac{\hat{t}_\mu}{\hat{c}_\mu}$ (i.e., using our estimate $\hat{c}_\mu$ of c_μ to estimate the denominator of a_μ) is not an unbiased estimator.

Shortcomings. This approach eliminates bias in query answering, so that the remaining error is due to the variance of the estimators (recall that error can be decomposed into bias and variance, see Eq. 1). Although this helps improve the accuracy for COUNT queries, in practice, we observed that, in the case of AVG queries, the large variance of the unbiased estimator leads to a larger error than the biased estimator. The difference in the effectiveness of debiasing for COUNT and AVG queries can be attributed to the difference in the estimator's variance. In our dataset, the number of visits of individuals has a much lower variance than the time people stay in different locations or their average distance traveled, as shown in Sec. V-C. Thus, debiasing does not help reduce error for average queries, as it does not reduce the variance which is the main source of error. Furthermore, the weights used for debiasing can often be large, further increasing the variance of the debiased estimator, leading to worse accuracy.

C. Learned Estimation

For average queries, the two approaches discussed so far show an example of bias/variance trade-off in estimation, where we see lowering the bias in our estimation increases the variance and leads to worse error. This shows that eliminating bias in our estimator can lead to worse results. Instead, we see that using an estimator with a correct inductive bias is able to provide more accurate results.

To provide lower error, we use a learned estimator to answer queries, where we train a model that uses information about a neighbourhood to estimate query answers. Intuitively, we use the inductive bias that there exist correlations between neighbourhood characteristics and query answers for the neighbourhood, and to reduce the error, a learned model is used to capture such correlations. For instance, similar POIs tend to have similar visit durations, and the model can aggregate such information across neighbourhoods to reduce error observed for each neighbourhood. Furthermore, such correlations between location characteristics and query answers can be learned from queries for which accurate answers are available. This takes advantage having more data available for some neighbourhoods. Therefore a model can learn accurate patterns from those neighbourhoods that can be extrapolated to neighbourhoods for which less data is available. We provide an overview of this approach in Sec. IV-C1 and describe further details in Secs. IV-C2 and IV-C3.

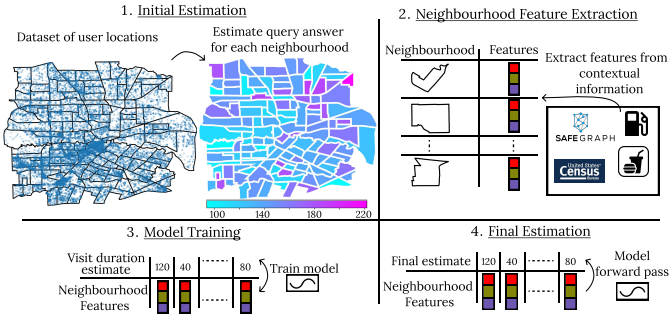

Figure 2: BiasBuster Overview

1) Overview: Figure 2 shows an overview of BiasBuster. BiasBuster has four steps as described below.

1. Initial Estimation. We use either the oblivious method of Sec. IV-A or the method using statistical debiasing of Sec. IV-B to obtain an initial estimate for the query answer for each neighbourhood. These initial estimates are later used by the model to obtain the final query answers estimate, where the training process extracts the underlying correlations from these initial estimates without overfitting to their error.

2. Neighbourhood Feature Extraction. We create features for each neighbourhood using contextual information available about the neighbourhood through auxiliary data sources. Intuitively, the feature vector captures characteristics of the neighbourhood that are relevant to the query answer, so that a model can learn the correlations between such characteristics and query answers. For instance, the average visit duration in a neighbourhood, is expected to be related to the type of POIs that exist in the neighbourhood. Thus, the neighbourhood features will contain information about types of POIs in the neighbourhood. This step is further described in Sec. IV-C2.

3. Model Training. Model training uses the initial estimates obtained in Step 1 and neighbourhood feature vectors obtained in Step 2 to learn a neural network through a supervised learning approach where the neighbourhood features are the input to the model and the model is trained to estimate query answers. This step is further described in Sec. IV-C3.

4. Final Estimation. The final estimates are obtained by performing a forward pass of the model for each neighbour-

hood. For each neighbourhood, the neighbourhood features from Step 2 is used, further described in Sec. IV-C3.

2) Neighbourhood Feature Extraction: We extract a set of features for each neighbourhood from auxiliary data sources.

POI Features. We utilize the information of POIs within the neighbourhood to characterise the neighbourhood. Specifically, we use the distribution of POI categories within each neighbourhood. We utilize Safegraph Places [36] (Safegraph Places [36] only provides a list of POIs, and is not generated based on user cellphone data), which contains list of POIs in all neighbourhoods, and for each POI the category it belongs to (e.g., if it's a restaurant or a hospital). The distribution of POI categories is a vector of length k, where k is the number of categories, whose i-th element is calculated by counting the number of POIs in the i-th category in the neighbourhood and dividing it by the total number of POIs in the neighbourhood.

Demographic features. We also use demographic features for each neighbourhood, specifically population and median income of the neighbourhood, both obtained from census.

3) Model Training and Inference: For a neighbourhood μ, let the query answer estimate obtained in Step 1 be $\hat{y}_\mu$ and let the features obtained in Step 2 be e_μ.

Training. The training set for our model is $T = \{(e_\mu, \hat{y}_\mu), \forall \mu\}$. We consider two variations of the training process. We use (1) unweighted loss, where the model, $\hat{f}$ is simply trained to predict $\hat{y}_\mu$, that is, to minimize $\sum_{(e_\mu,\hat{y}_\mu)\in T}(\hat{f}(e_\mu;\theta) - \hat{y}_\mu)^2$. Furthermore, (2) we train the model while giving more weight to neighbourhoods for which we are more confident about the estimate. Specifically, let s_μ be the number of samples observed in a neighbourhood μ and let $w_\mu = \frac{s_\mu}{\max_\mu s_\mu}$. We use the weighted loss function $\sum_{(e_\mu,\hat{y}_\mu)\in T} w_\mu(\hat{f}(e_\mu;\theta) - \hat{y}_\mu)^2$. Intuitively, using a weighted loss function, the model is trained to capture correlations from neighbourhoods where there are more observations and therefore our initial estimate is more accurate. We use fully connected neural networks.

Note that the training process uses labels $\hat{y}_\mu$ that are not the ground-truth answers to the queries for a neighbourhood, but instead, an initial estimate obtained from the observed database. The goal of the training process is to learn the underlying correlations of these query answers with respect to the neighbourhood characteristics, without overfitting to the error in the estimated answers. To ensure this, we use small models and early stopping for our training process. That is, we stop the training process before the model fully fits to the training data, as fully fitting to the training data means the model will have the same error as the initial estimate. Early stopping helps the model stop when it captures the correlations in the data but before it overfits to the training labels. This is further experimentally explored in Sec. V-D.

Inference. Obtaining the final estimate for a neighbourhood μ, is done by performing the forward pass $\hat{f}(e_\mu;\theta)$, where e_μ is the feature vector obtained in Step 2. We note that e_μ was in the training set. However, by training a model that captures query answer patterns without overfitting to the training labels, $\hat{f}(e_\mu;\theta)$ will be a better estimate of the query answer than the training label, $\hat{y}_\mu$. This is facilitated by weighting the training samples as well as early stopping, as explained above.

City	Observed Pop.	Sampling Ratio	# Stay-Points
Houston	94,355	0.04	1,002,389
Chicago	133,178	0.03	1,493,640
San Francisco	24,855	0.03	938,500
Tulsa	26,976	0.04	277,077
Fargo	6,246	0.04	92,029

Table II: Summary of dataset statistics

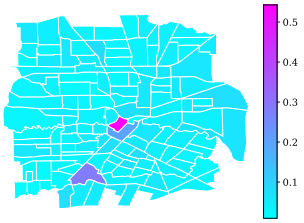

Figure 3: Sampling Ratios across Houston

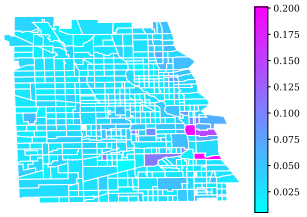

Figure 4: Sampling Ratios across Chicago

V. Empirical Study

A. Experiment Setup and Dataset Details

Our experiments use the dataset provided to us by Veraset [37], a data-as-a-service company that provides user location datasets. This dataset is the underlying dataset used by Safegraph [38], to provide aggregate population statistics, while also used by various other entities [39], [40], among them the US government [41]. We first describe this dataset and our preprocessing method in detail and then proceed to discuss the evaluation setup.

1) Dataset Details: We use the dataset provided to us for December 2019. The dataset consists of records of the form `user_id`, `latitude`, `longitude` and `timestamp`, where each record is obtained through phone GPS signals. From this dataset, we extract the stay-points sequences for users to obtain the dataset of the form described in Sec. III. We perform Stay Point Detection (SPD) [42] on the data to remove location signals when a person is moving, and to extract POI visits when a user is stationary.

Table II shows the details of our datasets after the above preprocessing steps. The sampling ratios for different cities reported in the third column of the table are calculated by finding the user's home neighbourhood and utilizing its corresponding Census tract demographics as the true population of a city, as described in Sec. III. Figs. 3 and 4 visualize that this sampling ratio is not distributed evenly across different neighbourhoods, and for some neighbourhoods our dataset contains data for a larger proportion of their population. Fig. 5 further quantifies this, showing how the sampling ratio varies across neighbourhoods.

To understand factors impacting the sampling ratio, we calculated its correlation between different demographic attributes of each neighbourhood. Specifically, we obtained the

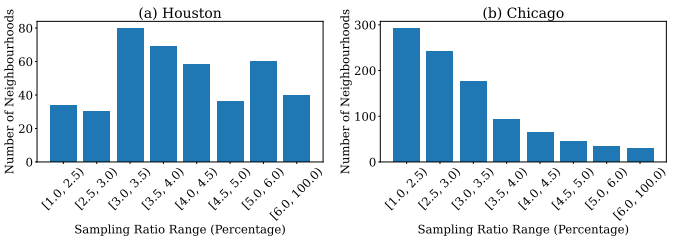

Figure 5: Sampling Ratios Across Neighbourhoods

median income for each neighbourhood from the US Census, and calculated the Pearson correlation coefficient between the median income of each neighbourhood and its sampling ratio. To calculate the correlation between age attributes, we obtained the population of a neighbourhood in an age range and divide it by the total population of that neighbourhood (both statistics obtained from census). Subsequently, we calculate the correlation coefficient between the calculated normalized age and sampling ratio for each neighbourhood. This normalization by total population is because we are interested in the correlation with sampling *ratio*, and not the size of the samples. These correlation statistics were reported in Table I, showing that, in our dataset, less data is available for seniors or children, while more data is available for the low-income population (we define the child age group as people below 18, adult as 18-65 and senior as above 65). Overall, the amount of data available depends on how the data was collected, the information that Veraset does not publically provide. However, we suspect that more data for low-income population is due to use of Android apps for data collection instead of iPhone apps, which are used more often by the higher-income population.

2) Experiment Setup:

Datasets. To evaluate our methods and to have access to a ground-truth, we consider the Veraset dataset, D, to be the true population and we sample a new subset, D_s from this dataset and consider it as the observed dataset. The sampling procedure is designed to mimic the true sampling procedure (based on which D was obtained). Specifically, we use the sampling ratios calculated using D (i.e., sampling ratios obtained by comparing D to the census data) for our sampling procedure. This ensures a similar sampling procedure to what was used to obtain D from the true population, is followed to obtain D_s from D. Specifically, for each neighbourhood μ whose sampling ratio is μ_s and whose population based on D is n_μ^D, we sample $\mu_s \times n_\mu^D$ users from users in D whose home neighbourhood is μ. This mimics the biased sampling process, where some neighbourhoods have a larger portion of their population observed. We sampled sampling uniformly within each neighbourhood from the population, which, as discussed before, can be true in practice since neighbourhoods themselves are often homogenous [34], [35].

In our experiments, we use datasets corresponding to multiple cities in the US, namely, Houston, Chicago, San Francisco, Tulsa and Fargo. The data for each city is extracted by defining an area of about 20x20 km^2 covering the city. For all algorithms, we sample the datasets five times and report the average error across the runs and its standard deviation.

Evaluation Metric. We evaluate our approach by considering three different estimation tasks, i.e., estimating the average visit duration, the average distance traveled and the total number of visits. Each estimate is for a different census tract (i.e., neighbourhood, see Sec. III) within a city and we report average relative error across the neighborhoods within a city. Specifically, let the true statistic (i.e., calculated from D) for a census tract μ from all tracts M be x_μ, and the estimate obtained from an algorithm (where the algorithm only has access D_s) be $\hat{x}_\mu$. We calculate the relative error over all census tracts as $\frac{1}{|M|}\sum_{\mu \in M} \frac{|x_\mu - \hat{x}_\mu|}{|x_\mu|}$.

Additionally, we also subdivide the census tracts into five categories and report average relative error for each category. The categories are defined based on how much of the data in each census tract was sampled. For each census tract, μ, let L_μ be the number of stay-points within the census tract and let l_μ be the number of stay-points sampled within that census tract. Then, we define the *stay-point sampling ratio* as $\frac{l_\mu}{L_\mu}$. Note that this is different from the *user sampling ratio*, defined in Sec. III, which considers how many of the users belonging to a neighbourhood were sampled. The user sampling ratio is used to describe the bias in sampling, which leads to different values for stay-point sampling ratios across the neighbourhood. On the other hand, stay-point sampling ratio is more correlated with the final error, since it is directly calculated based on the number of stay-point observations in a census tract. To understand the effect of stay-point sampling ratio, we divide the census tracts into five categories based on their stay-point sampling ratio. We consider the quantiles of sampling ratios within a city, and define the five categories as less than first quantiles, between first and second, between second and third, between third and fourth and more than fourth quantiles.

Methods. We present results for (1) oblivious estimation discussed in Sec. IV-A, referred to as Oblivious, (2) debiased estimation discussed in Sec. IV-B, referred to as Debiased and (3) variations of BiasBuster presented in Sec. IV-C. Specifically, we train BiasBuster with labels obtained from both Oblivious and Debiased, respectively referred to as BiasBuster-O and BiasBuster-D. Furthermore, we also train both these variations with weighted loss function which are referred to as BiasBuster-OW and BiasBuster-DW. All variations of BiasBuster are fully connected neural networks with 3 hidden layers and each layer of size 80.

B. Evaluation

1) Average visit duration: Table III and Fig. 6 show the results for this task. First consider Table III. It summarizes the error for different cities and different methods where for each column the number in parenthesis is the standard deviation of error across runs. First, we observe that across cities, all variations of BiasBuster outperform both Oblivious and Debiased significantly, reducing error by more than half in all instances. Second, Debiased performs worse than Oblivious across datasets, indeed showing that debiasing does not help improve accuracy as the large variance in estimation causes

City	Oblivious	Debiased	BiasBuster-O	BiasBuster-D	BiasBuster-OW	BiasBuster-DW
Houston	0.41 (±0.008)	0.43 (±0.007)	**0.20** (±0.007)	**0.20** (±0.004)	0.22 (±0.011)	0.21 (±0.008)
Chicago	0.49 (±0.013)	0.50 (±0.016)	0.24 (±0.008)	0.24 (±0.009)	**0.22** (±0.008)	**0.22** (±0.008)
San Francisco	0.43 (±0.028)	0.45 (±0.026)	0.21 (±0.019)	0.20 (±0.019)	**0.19** (±0.010)	**0.19** (±0.024)
Tulsa	0.45 (±0.020)	0.45 (±0.030)	0.25 (±0.026)	0.26 (±0.033)	0.24 (±0.027)	**0.23** (±0.022)
Fargo	0.34 (±0.032)	0.34 (±0.053)	0.25 (±0.027)	0.24 (±0.045)	**0.23** (±0.027)	0.24 (±0.050)

Table III: Relative Error for Average Visit Duration

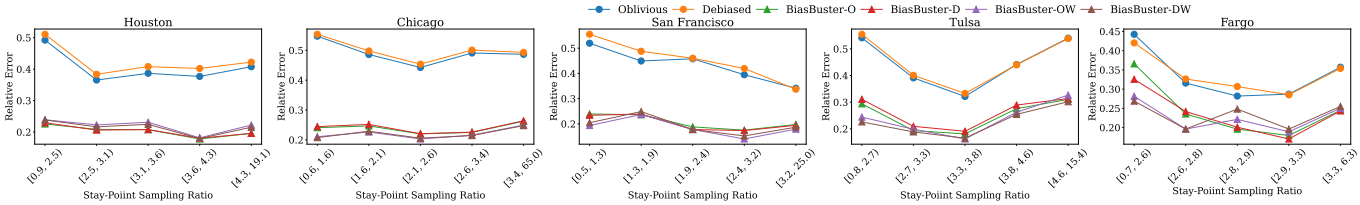

Figure 6: Error Across Different Sampling Ratios for Average Visit Duration Travelled

City	Oblivious	Debiased	BiasBuster-O	BiasBuster-D	BiasBuster-OW	BiasBuster-DW
Houston	0.49 (±0.023)	0.48 (±0.029)	0.41 (±0.053)	0.36 (±0.033)	0.27 (±0.015)	**0.23** (±0.012)
Chicago	0.73 (±0.026)	0.74 (±0.020)	0.44 (±0.154)	0.52 (±0.038)	**0.36** (±0.028)	0.38 (±0.032)
San Francisco	0.48 (±0.031)	0.50 (±0.019)	0.30 (±0.064)	0.26 (±0.035)	0.24 (±0.025)	**0.23** (±0.020)
Tulsa	0.46 (±0.070)	0.46 (±0.072)	0.38 (±0.037)	0.44 (±0.047)	**0.33** (±0.022)	**0.33** (±0.038)
Fargo	0.44 (±0.066)	0.45 (±0.091)	0.48 (±0.188)	0.51 (±0.107)	0.34 (±0.063)	**0.33** (±0.090)

Table IV: Relative Error for Average Distance Travelled

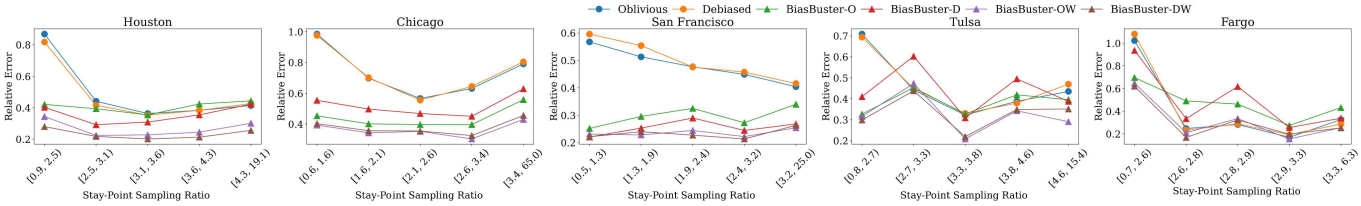

Figure 7: Error Across Different Sampling Ratios for Average Distance Travelled

large errors (we further discuss variance in estimation in Sec. V-C). Finally, we see that all variations of BiasBuster perform similarly, although the weighted variations, namely BiasBuster-OW and BiasBuster-DW often achieve better error (and/or lowest standard deviation), showing a marginal benefit for weighting the samples. We also note that although Debiased consistently performs worse than Oblivious, BiasBuster-D and BiasBuster-O (or similarly BiasBuster-DW and BiasBuster-OW) have a similar accuracy across datasets, showing that, using BiasBuster, it is possible to use worse training labels but still achieve good accuracy.

Furthermore, Fig. 6 further breaks down the error for different algorithms. We report the average error for different categories of census tracts defined based on stay-point sampling ratio of the census tracts (as described in Sec. V-A). Overall, we see that all variations of BiasBuster provide consistent accuracy across different sampling ratio, while for both Oblivious and Debiased, the error for census tracts with small sampling ratio is often more than 10% higher than the error for census tracts with higher sampling ratios. This shows that BiasBuster is able to provide consistent accuracy across neighbourhoods, even when their sampling ratio is small, thus avoiding penalizing communities for which less data is available (e.g., the seniors or children).

2) Average distance traveled: Table IV and Fig. 7 show the results for the task of estimating average distance traveled. The main observations are similar to the case of average visit duration. Table IV shows that Debiased does not perform better than Oblivious, while BiasBuster variations significantly outperform both. Moreover, Fig. 7 shows that BiasBuster provides consistently low accuracy across different sampling ratios, while both Oblivious and Debiased have very large variations in accuracy across sampling ratios (e.g, in Houston the error drops from 80% for locations with low sampling ratios to 40% for locations with high sampling ratios).

On the other hand, compared with average visit duration, for average distance traveled, we see that weighting the loss function has a more significant impact, where we often see more than 10% reduction in error when using the weighted loss. Overall, this shows that in the case of distance traveled, the model is able to better extrapolate the patterns from neighbourhoods with a large number of samples to neighbourhoods with a small number of samples. This can be because transferrable patterns from highly-sampled to less-sampled neighbourhoods are more prominent in the case of average distance traveled compared with average visit duration.

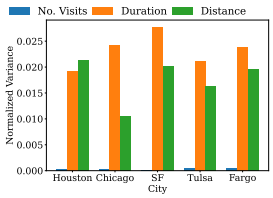

Figure 8: Variance of different attributes across users

City	Oblivious	Debiased	BiasBuster-D	BiasBuster-DW
Houston	0.70 (±0.052)	0.46 (±0.034)	**0.41** (±0.039)	0.46 (±0.072)
Chicago	1.07 (±0.076)	0.63 (±0.028)	**0.55** (±0.022)	0.64 (±0.049)
San Francisco	2.79 (±0.270)	0.63 (±0.069)	**0.59** (±0.074)	0.63 (±0.058)
Tulsa	0.81 (±0.043)	0.46 (±0.071)	**0.43** (±0.047)	0.49 (±0.082)
Fargo	0.98 (±0.208)	**0.41** (±0.074)	**0.41** (±0.043)	0.44 (±0.073)

Table V: Relative Error for Number of Visits

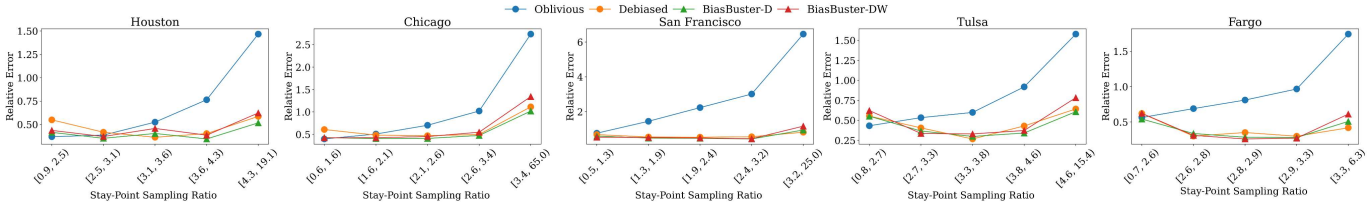

Figure 9: Error Across Different Sampling Ratios for Number of Visits

3) Total Number of Visits: Finally, Table V and Fig. 9 show the results for the task of estimating the total number of visits. In Table V, we see that in contrast to the two average statistics, Debiased does in fact improve upon Oblivious significantly. As shown later (Sec. V-C), this is due to the lower variance in number of visits of individuals, compared with average visit duration or distance traveled.Due to this significant difference betwe en Oblivious and Debiased, we only train BiasBuster using Debiased as its labels, so that we only report results for BiasBuster-D and BiasBuster-DW.

BiasBuster outperforms Debiased across all cities except Fargo, which can be because Fargo has the least number of neighbourhoods across all the cities. We also see that BiasBuster-D performs better than BiasBuster-DW. Recall that BiasBuster-DW weights neighbourhoods with more observed samples more heavily. In the case of average queries, this is helpful, as one expects to be able to learn patterns from such neighbourhoods and extrapolate to neighbourhoods with fewer samples. However, for total number of visits, such extrapolation is not as effective. This is because in the case of number of visits, neighbourhoods with more observed samples also tend to have more true visits. Therefore learning from such neighbourhoods leads to overestimating the number of visits for neighbourhoods where number of observed sampled is small. Finally, Fig. 9 shows Oblivious degrading significantly for neighbourhoods with large sampling ratios. This is because it scales the answer for all neighbourhoods with a fixed constant x but neighbourhoods with sampling ratios more than $\frac{1}{x}$ should be scaled with a smaller scaling factor.

C. Variance Analysis

In Sec. V-B, we saw that debiasing, compared with the oblivious approach, fails to improve accuracy in the case of `AVG` queries, while it does improve accuracy for the `COUNT` query. We discuss this further through the variance of the estimators which correlates with estimator error.

In Fig. 8, we plot the variance of the three attributes studied across different cities. To calculate the variance for the number of visits, we (1) for each neighbourhood calculate how many visits each user has in that neighbourhood, (2) calculate the variance of the number of visits across users for each neighbourhood and (3) present the average variance across all the neighbourhoods. For average visit duration and distance travelled, the process is the same, but in step (1), instead of calculating the number of visits of each user per neighbourhood, we calculate the average visit duration or distance travelled for the user in each neighbourhood. To be able to compare across the attributes, all the values are normalized to be in [0, 1] in step (2) and before calculating the variance, by deducting the minimum from all attributes and dividing them by the maximum value across the users.

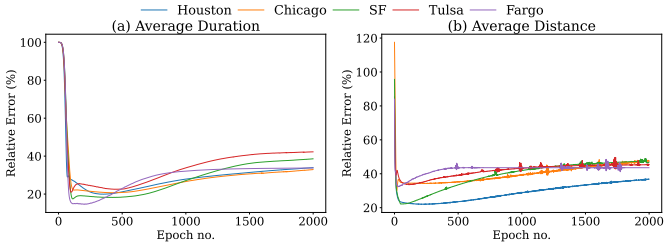

Figure 10: Test accuracy during training processing

Fig. 8 shows that both average visit duration and average distance traveled have a larger variance compared with the number of visits. This implies that the error in Oblivious for the average queries is more likely to be due to their variance, and not the bias; on the other hand the variance for the number of visits is very small (smaller than the average queries by orders of magnitude). Since debiasing only reduces the bias and not the variance, it does not help improve the accuracy for average queries (where variance is large), but it does improve accuracy for count queries (where variance is small).

D. Impact of Training Duration

Recall that the training process uses labels from the observed data (which are inaccurate), so it's prone to overfitting. Fig. 10 depicts test accuracy at different stages of training, which first improves but eventually worsens due to overfitting after 500 epochs. This shows that early stopping (i.e., stopping before the model fully fits the training set) is important for training accurate models.

VI. Conclusion

We presented the first comprehensive analysis of bias in real-world location data collected from mobile phones. We showed that this bias exists in a commonly used location dataset (used by Safegraph), that is often utilized for, among other tasks, the sensitive task of COVID forecasting and policy making. We showed that a statistical debiasing method often fails to improve accuracy, and instead presented BiasBuster, a neural network approach that utilizes correlations between location characteristics and population statistics to provide accurate estimates of population statistics. Our experiments showed that BiasBuster improves accuracy by up to 3 times for underrepresented populations. Future work includes extending location feature extraction to further improve accuracy and considering other types of aggregate queries.

Acknowledgements. This research has been funded in part by NSF grants IIS-2128661 and CNS-2125530 and NIH grant 5R01LM014026. Opinions, findings, conclusions, or recommendations expressed are those of the author(s) and do not necessarily reflect the views of any sponsors, such as NSF.

References

[1] S. Chang, E. Pierson, P. W. Koh, J. Gerardin, B. Redbird, D. Grusky, and J. Leskovec, "Mobility network models of covid-19 explain inequities and inform reopening," *Nature*, vol. 589, no. 7840, pp. 82–87, 2021.

[2] S. Zeighami, C. Shahabi, and J. Krumm, "Estimating spread of contact-based contagions in a population through sub-sampling," *Proceedings of the VLDB Endowment*, vol. 14, no. 9, pp. 1557–1569, 2021.

[3] S. Rambhatla, S. Zeighami, K. Shahabi, C. Shahabi, and Y. Liu, "Toward accurate spatiotemporal covid-19 risk scores using high-resolution real-world mobility data," *ACM TSAS*, vol. 8, no. 2, pp. 1–30, 2022.

[4] T. Hu, S. Wang, B. She, M. Zhang, X. Huang, Y. Cui, J. Khuri, Y. Hu, X. Fu, X. Wang *et al.*, "Human mobility data in the covid-19 pandemic: characteristics, applications, and challenges," *International Journal of Digital Earth*, vol. 14, no. 9, pp. 1126–1147, 2021.

[5] M. Haraguchi, A. Nishino, A. Kodaka, M. Allaire, U. Lall, L. Kuei-Hsien, K. Onda, K. Tsubouchi, and N. Kohtake, "Human mobility data and analysis for urban resilience: A systematic review," *Environment and Planning B: Urban Analytics and City Science*, 2022.

[6] Y. Song, G. Newman, X. Huang, and X. Ye, "Factors influencing long-term city park visitations for mid-sized us cities: A big data study using smartphone user mobility," *Sustainable Cities and Society*, 2022.

[7] K. Zhao, S. Tarkoma, S. Liu, and H. Vo, "Urban human mobility data mining: An overview," in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 1911–1920.

[8] C. Sun, P. Adamopoulos, A. Ghose, and X. Luo, "Predicting stages in omnichannel path to purchase: A deep learning model," *Information Systems Research*, vol. 33, no. 2, pp. 429–445, 2022.

[9] F. R. B. of Chicago, "Carts: Chicago fed advance retail trade summary," http://tinyurl.com/ycx5d6vb, 2023, accessed on 29th, May, 2023.

[10] H. Jin, S. Stubben, and K. Ton, "Customer shopping behavior and the persistence of revenues and earnings," *SSRN*, 2022.

[11] P. Newswire, "iphone users spend $101 every month on tech purchases, nearly double of android users," http://tinyurl.com/2snt3yun, accessed on 29th, May, 2023.

[12] netguru, "iphone vs android users: Key differences," http://tinyurl.com/45e38c7c, 2023, accessed on 29th, May, 2023.

[13] Hootsuite, "114 social media demographics that matter to marketers in 2023," http://tinyurl.com/ywdw28z9, accessed on 29th, May, 2023.

[14] S. Web, "View app demographics," http://tinyurl.com/4kpsvtzr, accessed on 29th, May, 2023.

[15] Safegraph, "Safegraph patterns documentation," https://docs.safegraph.com/docs/monthly-patterns, 2023, accessed on 29th, May, 2023.

[16] S. Lu, Z. Fang, X. Zhang, S.-L. Shaw, L. Yin, Z. Zhao, and X. Yang, "Understanding the representativeness of mobile phone location data in characterizing human mobility indicators," *ISPRS International Journal of Geo-Information*, vol. 6, no. 1, p. 7, 2017.

[17] A. Wesolowski, G. Stresman, N. Eagle, J. Stevenson, C. Owaga, E. Marube, T. Bousema, C. Drakeley, J. Cox, and C. O. Buckee, "Quantifying travel behavior for infectious disease research: a comparison of data from surveys and mobile phones," *Scientific reports*, 2014.

[18] S. Milusheva, D. Bjorkegren, and L. Viotti, "Assessing bias in smartphone mobility estimates in low income countries," in *ACM COMPASS*, 2021.

[19] A. Coston, N. Guha, D. Ouyang, L. Lu, A. Chouldechova, and D. E. Ho, "Leveraging administrative data for bias audits: assessing disparate coverage with mobility data for covid-19 policy," in *ACM FAccT*, 2021.

[20] A. Wesolowski, N. Eagle, A. M. Noor, R. W. Snow, and C. O. Buckee, "The impact of biases in mobile phone ownership on estimates of human mobility," *Journal of the Royal Society Interface*, vol. 10, no. 81, p. 20120986, 2013.

[21] G. Ranjan, H. Zang, Z.-L. Zhang, and J. Bolot, "Are call detail records biased for sampling human mobility?" *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 3, pp. 33–44, 2012.

[22] R. F. Squire, "What about bias in your dataset? quantifying sampling bias in safegraph patterns," https://tinyurl.com/3mketky6, 2019, accessed on 29th, May, 2023.

[23] Google, "Google mobility report," https://www.google.com/covid19/mobility/, 2022, accessed on 29th, May, 2023.

[24] Facebook, "Social connectedness index," http://tinyurl.com/yh42wts6, 2023, accessed on 29th, May, 2023.

[25] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, no. 1, 2017.

[26] K. Ouyang, R. Shokri, D. S. Rosenblum, and W. Yang, "A non-parametric generative model for human trajectories." in *IJCAI*, vol. 18, 2018, pp. 3812–3817.

[27] S. Zeighami, R. Seshadri, and C. Shahabi, "A neural database for answering aggregate queries on incomplete relational data," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[28] S. Zeighami and C. Shahabi, "Neurodb: Efficient, privacy-preserving and robust query answering with neural networks," in *NeurIPS 2023 Second Table Representation Learning Workshop*, 2023.

[29] S. Zeighami, R. Ahuja, G. Ghinita, and C. Shahabi, "A neural database for differentially private spatial range queries," *Proceedings of the VLDB Endowment*, vol. 15, no. 5, 2022.

[30] R. Ahuja, S. Zeighami, G. Ghinita, and C. Shahabi, "A neural approach to spatio-temporal data release with user-level differential privacy," *ACM SIGMOD*, 2023.

[31] F. R. Stevens, A. E. Gaughan, C. Linard, and A. J. Tatem, "Disaggregating census data for population mapping using random forests with remotely-sensed and ancillary data," *PloS one*, vol. 10, no. 2, p. e0107042, 2015.

[32] C. Robinson, F. Hohman, and B. Dilkina, "A deep learning approach for population estimation from satellite imagery," in *Proceedings of the 1st ACM SIGSPATIAL Workshop on Geospatial Humanities*, 2017, pp. 47–54.

[33] U. Census, "Census tracts and block numbering areas," http://tinyurl.com/2p42z8vs, accessed 5/2023.

[34] Brookings, "Even as metropolitan areas diversify, white americans still live in mostly white neighborhoods," http://tinyurl.com/mvdsvh9y, 2023, accessed on 29th, May, 2023.

[35] N. Y. Times, "They're rich but trying to reach beyond the money bubble," http://tinyurl.com/mrfc9bps, 2023, accessed on 29th, May, 2023.

[36] Safegraph, "Safegraph places documentation," https://docs.safegraph.com/docs/places#section-patterns, 2023, accessed on 29th, May, 2023.

[37] Veraset, "Veraset website," https://www.veraset.com/, accessed 5/2023.

[38] Quartz, "Meet the company helping scientists study covid-19 with your location data," http://tinyurl.com/26zn4axh, 2023, accessed 5/2023.

[39] E. F. Foundation, "Data broker veraset gave bulk device-level gps data to dc government," http://tinyurl.com/y9fyresa, 2023, accessed 5/2023.

[40] H. AI, "Omnisci welcomes safegraph and veraset to its data catalog, providing poi/gps data for commercial, business, government," http://tinyurl.com/yp6s72n2, 2023, accessed on 29th, May, 2023.

[41] W. Post, "Data broker shared billions of location records with district during pandemic," http://tinyurl.com/y9fyresa, 2021, accessed 5/2023.

[42] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie, "Mining individual life pattern based on location history," in *International conference on mobile data management: Systems, services and middleware*. IEEE, 2009.