A Fast and Accurate Method for Estimating People Flow from Spatiotemporal Population Data

Yasunori Akagi, Takuya Nishimura, Takeshi Kurashima, Hiroyuki Toda

NTT Service Evolution Laboratories, NTT Corporation,

1-1 Hikari-no-oka, Yokosuka-Shi, Kanagawa, 239-0847 Japan

{akagi.yasunori, nishimura.takuya,kurashima.takeshi,toda.hiroyuki }@lab.ntt.co.jp

Abstract

Real-time spatiotemporal population data is attracting a great deal of attention for understanding crowd movements in cities. The data is the aggregation of personal location information and consists of just areas and the number of people in each area at certain time instants. Accordingly, it does not explicitly represent crowd movement. This paper proposes a probabilistic model based on collective graphical models that can estimate crowd movement from spatiotemporal population data. There are two technical challenges: (i) poor estimation accuracy as the traditional approach means the model would have too many degrees of freedom, (ii) excessive computation cost. Our key idea for overcoming these two difficulties is to model the transition probability between grid cells (cells hereafter) in a geospatial grid space by using three factors: departure probability of cells, gathering score of cells, and geographical distance between cells. These advances enable us to reduce the degrees of freedom of the model appropriately and derive an efficient estimation algorithm. To evaluate the performance of our method, we conduct experiments using real-world spatiotemporal population data. The results confirm the effectiveness of our method, both in estimation accuracy and computation cost.

1 Introduction

Spatiotemporal population data are considered to be important in various fields, such as demand prediction for taxi operating systems [1], marketing [Hess *et al.*, 2004], and simulating human movements after massive disasters [Sudo *et al.*, 2016]. Spatiotemporal population data contains population snapshots of each area, and is created by aggregating personal location information obtained from mobile network operationing data [Terada *et al.*, 2013], GPS [Witayangkurn *et al.*, 2013] or Wi-Fi. This data format has the advantage that privacy is addressed while providing a sufficient amount of

---

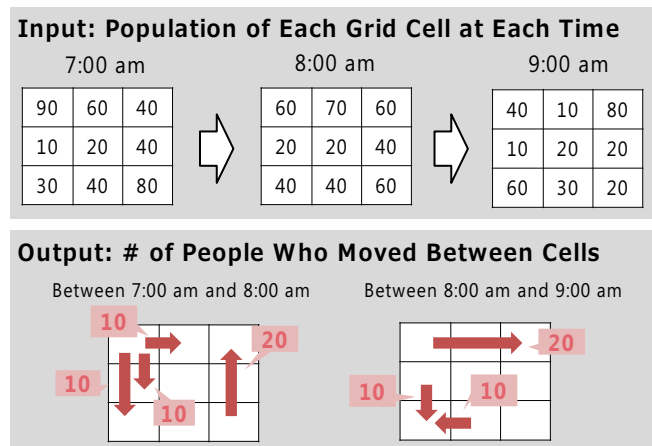[1]https://www.nttdocomo.co.jp/english/info/media_center/event/mwc2017/pdf/about_ai_taxi.pdf

Figure 1: Task description. We consider a grid space created by dividing the geospatial space into cells at equal intervals by lines parallel to latitude and longitude; this is a common space-discretization approach. The input is spatiotemporal population data, which is the number of people in each cell at each time step. The output is the number of people who moved between cells over time.

position information to understand macroscopic human behavior. Therefore, information sets using this data format are becoming popular. For example, *mobile spatial statistics* [Terada *et al.*, 2013] are calculated from mobile network operational data, the hourly population data of square grids (fixed size) in Japan. The XData project in France considered releasing hourly spatiotemporal population density data of each region of Paris [Acs and Castelluccia, 2014].

This paper tackles the task of estimating people flows from spatiotemporal population data as shown in Figure 1. Realizing this task will greatly extend the applicability of spatiotemporal population data, such as predicting taxi demand with consideration of flow direction, designing transportation systems tailored to mass movements, and detecting anomalies in the movement patterns of people.

One method for solving this task is based on the Collective Graphical Model (CGM) [Sheldon and Dietterich, 2011], which is a general framework for analyzing aggregated data. In this framework, the number of people who moved between cells is treated as a hidden variable, and this hidden variable

and transition probabilities between cells are estimated simultaneously. Iwata et al. applied CGM to the task of estimating people movement (just movement hereafter) from spatiotemporal population data, the same task as ours, and reported the effectiveness of their method [Iwata *et al.*, 2017]. However, their method is designed for estimating movements from one cell to neighboring cells. Thus the method cannot estimate movements over longer distances. When the discretization width of space (i.e. cell size) is small or time discretization width is large, many people move from one cell to non-neighboring cells between time steps. In such cases, the previous method offers poor estimation accuracy. In addition, since movement speeds vary widely with the mode of movement (walking, train, car, etc.), considering only movement to neighbor cells may undercount the number moving. This may yield wrong conclusions and hinder analysis.

At first sight, estimation that considers non-neighbor movement can be easily realized by preparing hidden variables and transition probability parameters to non-neighbor cells. However, this naive solution suffers from two crucial difficulties. (i) When the number of hidden variables and transition probability parameters is large, too many solutions are consistent with the observed population data. As a result, the estimation process readily falls into local optima, resulting in low estimation accuracy. (ii) As the number of hidden variables increases, the computation cost of updating the hidden variables becomes excessive. Since we have to update hidden variables iteratively to estimate them, the total algorithm is not really scalable.

This paper proposes a new model (based on CGM) and its inference algorithm for estimating people flows from spatiotemporal population data. Our key idea for overcoming the two difficulties described above is to model the transition probability from cell i to cell j by using three factors; (i) departure probability of i which represents how likely people at i are to move to other cells, (ii) gathering score of j which represents how likely people are to gather at cell j, and (iii) the geographical distance between i and j. This model enables us to appropriately reduce the degrees of freedom in the parameter inference procedure. As a result, estimation becomes less likely to be trapped in local optima which makes highly accurate estimates possible. Moreover, this model enables us to derive an approximate scalable inference algorithm; by utilizing sufficient statistics of parameters in this model, it is possible to drastically reduce the hidden variables that need to be processed in the algorithm, resulting in much lighter computation cost. This paper proposes both exact and approximate estimation algorithms and compares them.

To evaluate the performance of our method, we conduct experiments on a real-world spatiotemporal population data set created by aggregating GPS trace data of cars obtained via car navigation systems. The results confirmed the high estimation accuracy of our method and light computation cost of the proposed approximate estimation algorithm.

2 Problem Formulation

We consider a grid space created by dividing the geospatial space at equal intervals by lines parallel to latitude and

Symbol	Definition
V	set of all cells
T	number of timesteps
N_{ti}	population of cell i at time step t
M_{tij}	# of people who moved from cell i to cell j between time step t and $t+1$
θ_{ij}	transition probability from cell i to cell j
Γ_i	set of destination candidate cells for transition from cell i
K	parameter for controlling the size of Γ_i
$d(i,j)$	distance between cell i and cell j
π_i	departure probability of cell i which represents how likely people at cell i are to move to cells other than cell i
s_i	gathering score of cell i which represents how likely people are to gather at cell i
β	parameter representing transition probability decay (depends on distance between cells)
λ	hyperparameter for controlling the penalty

Table 1: Notation

longitude. The problem we tackle with in this paper is formulated as follows. Suppose we have population of cell i at timestep t, $N_{ti}(i \in V, t = 0, \ldots, T-1)$, where V is the set of all cells in the grid space and T is the number of timesteps. Our goal is to estimate the population who leave cell i at time t and whose next cell is j at time $t+1$, $M_{tij}(i \in V, j \in V, t = 0, \ldots, T-2)$. Note that, we can apply our method to spatiotemporal population data discretized in other ways; all that is required is to design an appropriate distance function between discretized areas.

The notations used in this paper are listed in Table 1.

3 Related Work

3.1 Collective Graphical Model

Collective Graphical Model (CGM) [Sheldon and Dietterich, 2011] is a general framework for inferring the hidden probabilistic model underlying a set of aggregated data. This framework has been applied to the analysis of transition patterns of vehicles in road networks [Sheldon *et al.*, 2013], and pedestrian data analysis for a large amusement park [Du *et al.*, 2014]. Of particular interest, Iwata et al. deal with the problem of estimating transition population from the population data of each cell, which is the same task we tackle in this paper [Iwata *et al.*, 2017]. The relationship of these methods and our proposed method is explained in Section 4.1.

Various estimation techniques for CGM have been published, such as Gibbs sampler method [Sheldon and Dietterich, 2011], the message passing based method [Sun *et al.*, 2015], and the MAP-based estimation method [Sheldon *et al.*, 2013]. Our estimation scheme is based on MAP-based estimation [Sheldon *et al.*, 2013].

3.2 Transition Estimation From Aggregated Data

Kumar et al. deal with the problem of inferring transition tendencies between items (e.g. web pages) based on aggre-

gate data containing the relative popularity of items (e.g. the number of views of web pages) [Kumar *et al.*, 2015]. They formulate this problem as reconstructing the transition matrix of a Markov chain from an observed stationary distribution. Although this problem is ill-posed and has infinitely many solutions in general, they showed that a unique solution can be determined using Luce's Choice model [Luce, 1959]. Maystre et al. extended this approach and proposed an algorithm named Choice Rank for reconstructing the transition matrix of a Markov chain on a graph from the incoming and outgoing population of each vertex under Luce's choice model [Maystre and Grossglauser, 2017]. These methods are not designed to treat time-discretized population data, and so cannot be applied to our problem setting.

Xu et al. propose an algorithm for restoring personal movement histories from spatiotemporal population data in order to evaluate the privacy risk of publishing spatiotemporal population data [Xu *et al.*, 2017]. Their method solves the linear sum assignment problem under cost functions decided via an analysis of real mobility data. Although this method is similar to methods based on CGM (including ours) in terms of solving optimization problem under cost functions, this method has no mechanism to estimate the parameters (i.e. transition probabilities between cells) that determine the cost functions. Therefore, the data to which this method can be applied is limited. Our method estimate people's movements without heuristic assignment of transition probabilities between cells.

3.3 Future Prediction of Population

A lot of studies have attempted to predict future population from past population data or other features. For example, Hoang et al. tackle the problem of predicting incoming and outgoing population of each area in a city and proposed a prediction method based on decomposing dynamics into seasons, social trends, and other components [Hoang *et al.*, 2016]. Zhang et al. treated the same problem in proposing ST-ResNet, which is a deep-learning-based method [Zhang *et al.*, 2017]. They are supervised methods and so require both input (e.g. past incoming population) and output variables (e.g. future incoming population) to train their predictors. On the other hand, our proposed method is an unsupervised learning method and so needs only input variables (spatiotemporal population data) and does not need output variables (transition population between cells).

4 Proposed Method

4.1 Preliminary

We first explain the method proposed by Iwata et al. [Iwata *et al.*, 2017] that uses CGM to estimate transition populations from the observed number of people in each cell.

Let Γ_i be a set of candidate cells that can be destinations of movements from cell i. In previous work, Γ_i is a set of cells within L_∞-distance 1 from cell i, since the method is designed for estimating movements from one cell to its neighbors. Given population N_{ti} and transition probability $\boldsymbol{\theta}_i = \{\theta_{ij}\}_{j \in \Gamma_i}$, the transition population $\boldsymbol{M}_{ti} = \{M_{tij}\}_{j \in \Gamma_i}$ ($t = 0, 1, \ldots, T-2, i \in V$) given population N_{ti}

is assumed to be decided by the following multinomial distribution: $P(\boldsymbol{M}_{ti} \mid N_{ti}, \boldsymbol{\theta}_i) = \frac{N_{ti}!}{\prod_{j \in \Gamma_i} M_{tij}!} \prod_{j \in \Gamma_i} \theta_{ij}^{M_{tij}}$. Given $\boldsymbol{N} = \{\boldsymbol{N}_{ti} \mid t = 0, \ldots, T-1, i \in V\}$ and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_i \mid i \in V\}$, the likelihood function of $\boldsymbol{M} = \{\boldsymbol{M}_{ti} \mid t = 0, \ldots, T-2, i \in V\}$ is calculated as

$$P(\boldsymbol{M} \mid \boldsymbol{N}, \boldsymbol{\theta}) = \prod_{t=0}^{T-2} \prod_{i \in V} \left(\frac{N_{ti}!}{\prod_{j \in \Gamma_i} M_{tij}!} \prod_{j \in \Gamma_i} \theta_{ij}^{M_{tij}} \right).$$

The population in cell N_{ti} and the transition population between cells $\boldsymbol{M}_{ti}$ satisfy the following two relationships, they represent the law of conservation in the number of people: $N_{ti} = \sum_{j \in \Gamma_i} M_{tij}, N_{t+1,i} = \sum_{j \in \Gamma_i} M_{tji}$ ($t = 0, 1, \ldots, T-2$). Estimation is done by maximizing the log-likelihood function $\log P(\boldsymbol{M} \mid \boldsymbol{N}, \boldsymbol{\theta})$ under constraints of the law of conservation described above. These constraints might not be satisfied strictly in real-world datasets, because observations are always noisy and obscuration may be used to preserve user privacy. To deal with these problems, these constraints are treated as soft constraints and we try to minimize the penalty term defined as below:

$$\mathcal{C}(\boldsymbol{M}) \equiv \sum_{t=0}^{T-2} \left| N_{ti} - \sum_{j \in \Gamma_i} M_{tij} \right|^2 + \sum_{t=0}^{T-2} \left| N_{t+1,i} - \sum_{j \in \Gamma_i} M_{tji} \right|^2.$$

It follows that objective function $\mathcal{L}(\boldsymbol{M}, \boldsymbol{\theta})$ is given by

$$\mathcal{L}(\boldsymbol{M}, \boldsymbol{\theta}) \equiv \log P(\boldsymbol{M} \mid \boldsymbol{N}, \boldsymbol{\theta}) - \frac{\lambda}{2} \cdot \mathcal{C}(\boldsymbol{M})$$
$$\approx \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in \Gamma_i} (M_{tij} \log \theta_{ij} + M_{tij} - M_{tij} \log M_{tij})$$
$$- \frac{\lambda}{2} \cdot \mathcal{C}(\boldsymbol{M}), \tag{1}$$

where constants that do not depend on $\boldsymbol{M}, \boldsymbol{\theta}$ are omitted, and we apply Stirling's approximation to the $\log M_{tij}!$ term, in order to calculate $\log M_{tij}!$ efficiently [Sheldon *et al.*, 2013]. λ is a hyperparameter that controls the penalty. Optimization is done by updating $\boldsymbol{M}$ and $\boldsymbol{\theta}$ alternately. This optimization process can be interpreted as the EM algorithm whose E-step is approximately calculated [Sheldon *et al.*, 2013].

4.2 Technical Challenges and Main Ideas

The previous method described in the previous section is designed for estimating movements from one cell to its neighboring cells, and so cannot estimate movements over longer distances. At first sight, estimation that considers non-neighbor movement can be easily realized by preparing hidden variables and transition probability parameters to non-neighbor cells. However, there are two technical challenges to estimate the number of people who moved between cells including non-neighbor movements: (i) As the number of hidden variables $\boldsymbol{M}$ and transition probability parameters $\boldsymbol{\theta}$ increases, the degree of freedom of the model becomes too high (i.e., too many solutions are consistent with the observations). As a result, estimation is easily trapped in local

Algorithm 1 Exact Inference Algorithm

1: **Input**: population data: $\boldsymbol{N}$
2: **Output**: movement $\boldsymbol{M}$, parameter $\boldsymbol{\pi}, \boldsymbol{s}, \beta$
3: Initialize parameters $\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta$
4: **while** condition $\mathcal{L}'$ **do**
5: $\quad \boldsymbol{M} \leftarrow \arg\max\limits_{\boldsymbol{M}} \mathcal{L}'(\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta)$
6: $\quad \boldsymbol{\pi} \leftarrow \arg\max\limits_{\boldsymbol{\pi}} \mathcal{L}'(\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta)$
7: $\quad \boldsymbol{s}, \beta \leftarrow \arg\max\limits_{\boldsymbol{s}, \beta} \mathcal{L}'(\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta)$
8: **end while**
9: **return** $\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta$

optima, resulting in output $\boldsymbol{M}$ far from the correct answer. (ii) As the number of $\boldsymbol{M}$ elements increases, the computation cost of updating $\boldsymbol{M}$ becomes very heavy. Since we have to conduct update iteratively, the total algorithm becomes non-scalable.

Our key idea to overcome the two difficulties described above is using three factors to describe the transition probability from cell i to cell j: departure probability of i, gathering score of j, distance between i and j. Adopting this model makes it possible to appropriately reduce the degree of freedom of the model while considering movement to non-neighbor cells. As a result, estimation becomes less likely to be trapped in bad local optima and highly accurate estimation becomes possible. Moreover, this model enables us to derive an approximate scalable inference algorithm; by utilizing sufficient statistics of parameters in this model, it is possible to drastically reduce the hidden variables that need to be processed in the algorithm, resulting in much lighter computation cost.

4.3 Proposed Model

Modeling Transition Probability. In our model, we set Γ_i to be a set of cells within distance K from cell i (i.e. $\Gamma_i = \{j \mid j \in V, d(i, j) \leq K\}$, where $d(i, j)$ is the distance between i and j). As a distance function, we use L_i-distance ($i = 1, 2, \infty$, etc) between cells. We assume that the transition probability from cell i to cell j can be written as

$$\theta_{ij} = \begin{cases} 1 - \pi_i & (i = j) \\ \pi_i \dfrac{s_j \cdot e^{-\beta \cdot d(i,j)}}{\sum_{k \in \Gamma_i \setminus \{i\}} s_k \cdot e^{-\beta \cdot d(i,k)}} & (j \in \Gamma_i \setminus \{i\}), \\ 0 & (\text{o.w.}) \end{cases} \quad (2)$$

where π_i is departure probability, which is the probability of moving to cells other than the current timestep cell. $s_j > 0$ is a value that represents how likely people are to gather at cell j. The exponential decay function $e^{-\beta \cdot d(i,j)}$ determines how likely cell j is to be chosen after current timestep considering the distance between cell i and cell j. Parameter β describes the shape of the decay function.

We provide an intuitive explanation of the transition probability modeling method (2). The next cell of a person who was at cell i is decided as follows:

- First, probability π_i determines whether the person leaves cell i or stays.

- If the person is deemed to leave cell i, the next cell is decided according to the probability proportional to the score which is determined for each candidate cell $j \in \Gamma_i \setminus \{i\}$. The score represents how likely cell j is to be chosen after i, and is calculated by multiplying s_j by $e^{-\beta \cdot d(i,j)}$.

For example, in the morning hours, cell j, which includes the business district, is expected to have high gathering score s_j since many people head to work in cell j, while cell i, which includes a residential area, is expected to have high departure probability π_i. In the evening hours, the converse is expected to be true. Moreover, the next cell that a person will move to largely depends on the current cell. For example, a person who lives in Oakland is more likely to commute to Union Square in San Francisco than Manhattan in New York City because Union Square is much closer. Although our distance factor is based on an exponential function for simplicity, other functions such as power function can be used as the distance factor.

Log-Likelihood Function. By substituting (2) into (1), we get the following new objective function

$$\mathcal{L}'(\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta) \equiv \sum_{t=0}^{T-2} \sum_{i \in V} \log(1 - \pi_i) M_{tii}$$

$$+ \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in \Gamma_i \setminus \{i\}} \left[\left\{ \log \pi_i + \log s_j - \beta \cdot d(i,j) \right. \right.$$

$$\left. \left. - \log \sum_{k \in \Gamma_i \setminus \{i\}} s_k \cdot e^{-\beta \cdot d(i,k)} \right\} \cdot M_{tij} \right]$$

$$+ \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in \Gamma_i} \left(M_{tij} - M_{tij} \log M_{tij} \right) - \frac{\lambda}{2} \cdot \mathcal{C}(\boldsymbol{M}), \quad (3)$$

where constants are omitted.

4.4 Exact Inference

We present an exact inference algorithm for the proposed model. We maximize $\mathcal{L}'(\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta)$ by optimizing each parameter, $\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}$, and β, alternately, as shown below. The proposed algorithm is summarized in Algorithm 1.

Update of $\boldsymbol{M}$. Given the current estimates $\hat{\pi}, \hat{s}, \hat{\beta}$, the optimization problem for $\boldsymbol{M}$ is given by

$$\begin{aligned} \text{Max.} \quad & \mathcal{L}'(\boldsymbol{M}, \hat{\boldsymbol{\pi}}, \hat{\boldsymbol{s}}, \hat{\beta}), \\ \text{s.t.} \quad & M_{tij} \geq 0 \quad (t = 0, \ldots, T-2, i \in V, j \in \Gamma_i). \end{aligned} \quad (4)$$

Because the objective function is concave for $\boldsymbol{M}$, we can get a global optimum by gradient-based methods such as the L-BFGS-B method [Byrd *et al.*, 1995].

Update of $\boldsymbol{\pi}$. Given the current estimates $\hat{\boldsymbol{M}}$, a closed form maximizer of the objective function (3) under constraints $0 \leq \pi_i \leq 1$ ($i \in V$) can be obtained by using the Lagrangian multiplier method:

$$\pi_i = \frac{\sum_{t=0}^{T-2} \sum_{j \in \Gamma_i \setminus \{i\}} \hat{M}_{tij}}{\sum_{t=0}^{T-2} \sum_{j \in \Gamma_i} \hat{M}_{tij}}. \quad (5)$$

Update of s, β. By extracting the terms dependent on s, β from objective function (3), we get

$$\sum_{i \in V} \left[A_i \log s_i - B_i \log \left(\sum_{k \in \Gamma_i \setminus \{i\}} s_k \cdot e^{-\beta \cdot d(i,k)} \right) \right] - \beta D, \tag{6}$$

where
$A_i \equiv \sum_{t=0}^{T-2} \sum_{j \in \Gamma_i \setminus \{i\}} \hat{M}_{tji}$, $B_i \equiv \sum_{t=0}^{T-2} \sum_{j \in \Gamma_i \setminus \{i\}} \hat{M}_{tij}$,
$D \equiv \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in \Gamma_i \setminus \{i\}} d(i,j) \cdot \hat{M}_{tij}$. We denote (6) by $f(s, \beta)$.

In order to maximize $f(s, \beta)$, we utilize the framework called the Minorization-Maximization algorithm (MM algorithm) [Hunter, 2003]. MM algorithm is an optimization technique that generates series of solution candidates by maximizing the lower bound of the objective function sequentially, instead of maximizing the objective function directly.

We explain the maximization procedure in detail here. Using the inequality

$$-\log x \geq 1 - \log y - \frac{x}{y} \quad (x, y > 0) \tag{7}$$

for

$$x_i = \sum_{k \in \Gamma_i \setminus \{i\}} s_k \exp(-\beta \cdot d(i,k)),$$

$$y_i = \sum_{k \in \Gamma_i \setminus \{i\}} s_k^{(u)} \exp(-\beta^{(u)} \cdot d(i,k)),$$

we get

$$f(s, \beta) \geq f^{(u)}(s, \beta) \tag{8}$$

with

$$f^{(u)}(s, \beta) \equiv$$
$$\sum_{i \in V} \left[A_i \log s_i - C_i^{(u)} \sum_{k \in \Gamma_i \setminus \{i\}} s_k \exp(-\beta \cdot d(i,k)) \right] - \beta D,$$
$$C_i^{(u)} \equiv \frac{B_i}{\sum_{k \in \Gamma_i \setminus \{i\}} s_k^{(u)} \exp(-\beta^{(u)} \cdot d(i,k))}.$$

In other words, $f^{(u)}(s, \beta)$ is a lower bound of $f(s, \beta)$. In the proposed algorithm, we maximize $f(s, \beta)$ by maximizing this lower bound function iteratively. An overview of the proposed algorithm is given in Algorithm 2

Update of s in Algorithm 2 can be done in closed form as follows

$$s_i = \frac{A_i}{\sum_{k \in \Gamma_i \setminus \{i\}} C_k^{(u)} \exp \left(-\beta^{(u)} \cdot d(k,i) \right)}.$$

While β in Algorithm 2 cannot be updated in closed form, we can get $\beta^{(u+1)}$ efficiently using golden section search [Kiefer, 1953] or Newton's method [Boyd and Vandenberghe, 2004] because $f^{(u)}(s^{(u+1)}, \beta)$ is concave for β.

Because equality holds in (7) when $x = y$,

$$f(s^{(u)}, \beta^{(u)}) = f^{(u)}(s^{(u)}, \beta^{(u)}) \tag{9}$$

Algorithm 2 Maximization of $f(s, \beta)$

1: Initialize $s^{(0)}, \beta^{(0)}$
2: $u \leftarrow 0$
3: **while** improving f **do**
4:     $s^{(u+1)} \leftarrow \arg \max_{s} f^{(u)}(s, \beta^{(u)})$
5:     $\beta^{(u+1)} \leftarrow \arg \max_{\beta} f^{(u)}(s^{(u+1)}, \beta)$
6:     $u \leftarrow u + 1$
7: **end while**
8: **return** $s^{(u)}, \beta^{(u)}$

---

holds. Using (8)(9), we get

$$f(s^{(u+1)}, \beta^{(u+1)}) \geq f^{(u)}(s^{(u+1)}, \beta^{(u+1)}) \quad (\because (8))$$
$$\geq f^{(u)}(s^{(u+1)}, \beta^{(u)}) \geq f^{(u)}(s^{(u)}, \beta^{(u)})$$
$$= f(s^{(u)}, \beta^{(u)}) \quad (\because (9)).$$

This means it is guaranteed that the objective function $f(s, \beta)$ montonically increases in Algorithm 2.

4.5 Approximate Inference

The computation bottleneck of the exact inference (Algorithm 1) is the update of M, since we have to solve a convex optimization problem with a large number of variables. For example, if we use an L_2-distance function, the number of variables to be optimized is $(T - 1) \cdot \left(\sum_{i \in V} |\Gamma_i| \right) \approx O(|V|K^2T)$. Therefore, if K is large, that is to say we consider large movements, the number of variables becomes excessive. Because we have to solve this convex optimization problem iteratively, computation cost becomes excessive.

In order to avoid this, we propose here an efficient approximate estimation algorithm that has significantly fewer variables. The key observation is as follows: if we set

$$X_{ti\delta} \equiv \sum_{j \in \Gamma_{i\delta}} M_{tji}, \ Y_{ti} \equiv \sum_{j \in \Gamma_i \setminus \{i\}} M_{tij}, \ Z_{ti} \equiv M_{tii},$$

where $\Gamma_{i\delta} \equiv \{j \in \Gamma_i \mid d(i,j) = \delta\}$ $(i \in V, \delta \in \Delta)$ and $\Delta \equiv \{r \mid \exists i \in V, \exists j \in \Gamma_i \setminus \{i\}, d(i,j) = r\}$, we can rewrite two Eqs.(5) (6) as follows:

$$\pi_i = \frac{\sum_{t=0}^{T-2} Y_{ti}}{\sum_{t=0}^{T-2} Y_{ti} + \sum_{t=0}^{T-2} Z_{ti}}, \tag{10}$$

$$f(s, \beta) = \sum_{i \in V} \left[\left(\sum_{t=0}^{T-2} \sum_{\delta \in \Delta} X_{ti\delta} \right) \log s_i \right.$$
$$\left. - \left(\sum_{t=0}^{T-2} Y_{ti} \right) \log \left(\sum_{k \in \Gamma_i \setminus \{i\}} s_k \exp(-\beta \cdot d(i,k)) \right) \right]$$
$$- \beta \cdot \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{\delta \in \Delta} \delta \cdot X_{ti\delta}. \tag{11}$$

These two equations imply that π, s, β can be updated by using only X, Y, Z (i.e. X, Y, Z are sufficient statistics of π, s, β), and M is not necessary. Based on this observation, we propose here an estimation algorithm that holds

Algorithm 3 Approximate Inference Algorithm

1: **Input**: population data: $\boldsymbol{N}$
2: **Output**: movement $\boldsymbol{M}$, parameter $\boldsymbol{\pi}, \boldsymbol{s}, \beta$
3: Initialize parameters $\boldsymbol{M}, \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{s}, \beta$
4: **while** condition $\mathcal{L}_{\text{approx}}$ **do**
5:     $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \leftarrow \underset{\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}}{\arg \max} \, \mathcal{L}_{\text{approx}}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{s}, \beta)$
6:     Update $\boldsymbol{\pi}$ by (10)
7:     Update $\boldsymbol{s}, \beta$ by maximizing $f(\boldsymbol{s}, \beta)$ calculated by (11)
8: **end while**
9: Final calculation of $\boldsymbol{M}$
10: **return** $\boldsymbol{M}, \boldsymbol{\pi}, \boldsymbol{s}, \beta$

$\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ instead of $\boldsymbol{M}$. Since the total number of elements of $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ is about $O(|V||\Delta|T)$, which is significantly less than the number of variables to be optimized in exact inference, $O(|V|K^2T)$, updating $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ is dramatically faster than updating $\boldsymbol{M}$. Since Δ is a set consisting of distance values between all pairs of cells, $|\Delta|$ varies depending on the distance function we adopt in the algorithm. To make $|\Delta|$ small, we should use an integer-valued distance function, such as L_1 or L_∞-distance function. The proposed algorithm is summarized in Algorithm 3.

Update of $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$. In order to update $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$, we have to minimize the objective function (1), where θ is calculated using the current parameter $\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{s}}, \hat{\beta}$ by Eq.(2). However, since $\boldsymbol{M}$ is not held in the algorithm, we cannot calculate the objective function (1) exactly. To overcome this issue, we minimize the approximate objective function of (1). We approximate M_{tij}, which is generated from binomial distribution $\text{Bin}(N_{tj}, \theta_{ij})$, by independent Poisson distribution $\text{Po}(N_{ti} \cdot \theta_{ij})$. Then, by utilizing the reproductive property of Poisson distributions, $X_{ti\delta} = \sum_{j \in \Gamma_{i\delta}} M_{tji}$ can be approximated as being generated from $\text{Po}(\mu_{i\delta})$ where $\mu_{i\delta} \equiv \sum_{j \in \Gamma_{i\delta}} N_{tj} \cdot \theta_{ji}$. Similarly, we generate Y_{ti} from $\text{Po}(N_{ti}\pi_i)$ and Z_{ti} from $\text{Po}(N_{ti}(1-\pi_i))$. Using these approximations, we get the following approximated log-likelihood function:

$$\mathcal{L}_{\text{approx}}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$$
$$\equiv \sum_{t=0}^{T-2} \sum_{i \in V} \sum_{\delta \in \Delta} \{X_{ti\delta} \log \mu_{i\delta} + X_{ti\delta} - X_{ti\delta} \log X_{ti\delta}\}$$
$$+ \sum_{t=0}^{T-2} \sum_{i \in V} \{Y_{ti} \log (N_{ti}\pi_i) + Y_{ti} - Y_{ti} \log Y_{ti}\}$$
$$+ \sum_{t=0}^{T-2} \sum_{i \in V} \{Z_{ti} \log (N_{ti}(1-\pi_i)) + Z_{ti} - Z_{ti} \log Z_{ti}\}.$$

We maximize $\mathcal{L}_{\text{approx}}$ under the constraint of the conservation of the number of people

$$N_{ti} = Y_{ti} + Z_{ti} \quad (t = 0, \dots, T-2, i \in V),$$
$$N_{t+1,i} = \sum_{\delta \in \Delta} X_{ti\delta} + Z_{ti} \quad (t = 0, \dots, T-2, i \in V).$$

We treat these constraints as penalty terms as we did in Section 4.1 and maximize the sum of $\mathcal{L}_{\text{approx}}$ and the penalty

terms under constraints $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \geq 0$. Because the objective function is concave for $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$, we can get a global optimum by gradient-based methods such as the L-BFGS-B method [Byrd *et al.*, 1995].

Update of $\boldsymbol{\pi}, \boldsymbol{s}, \beta$. We can update $\boldsymbol{\pi}, \boldsymbol{s}, \beta$ in the same way as explained in Section 4.4 using Eqs. (10) (11).

Final Calculation of $\boldsymbol{M}$. After iteratively updating $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}, \boldsymbol{\pi}, \boldsymbol{s}, \beta$, we calculate the estimated value of $\boldsymbol{M}$, which is what is needed to solve our problem. To do this, we solve the optimization problem (4) using $\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{s}}, \hat{\beta}$, which are the parameter values estimated by the iteration.

5 Experiments

We conducted experiments using real-world datasets to evaluate the estimation performance of our model; we estimated the number of people who moved between cells.

5.1 Settings

The datasets used in our experiments were generated from GPS trace data of cars in the Greater Tokyo Area collected by a car navigation application [2]. This car data contains trajectory data of each car, where each entry is a tuple of latitude, longitude, and time. We then calculated spatiotemporal cell populations from the trajectories, where the time interval is 60 minutes (from 6:00 to 21:00 at 1 hour intervals) and grid size was 10km × 10km or 5km × 5km. The number of unique users present in the datasets was 8694, and the number of cells was $8 \times 7 = 56$ for the 10km × 10km grid space and $16 \times 13 = 208$ for the 5km × 5km grid space. We also obtained the true values of the number of people who moved between cells from the car data in order to evaluate how close the outputs of our models were to the true values.

We evaluated the estimation accuracy using normalized absolute error (NAE), which is defined as

$$\frac{\sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in V} |M_{tij}^* - M_{tij}|}{\sum_{t=0}^{T-2} \sum_{i \in V} \sum_{j \in V} M_{tij}^*},$$

where M_{tij}^* is the true value of the number of people who moved from i to j at timestep t, and M_{tij} is the estimated value. Because the estimated values are real-valued, we round them (i.e., convert them into integer values).

The entire experimental procedure is explained as follows: 1) We calculated spatiotemporal cell populations N_{ti} from the original car trajectories. 2) Also, from the original trajectories, we calculated the true values of the number of people who moved between cells M_{tij}^*. 3) Given N_{ti}, we ran the proposed and baseline methods and got the estimated number of people who moved between cells M_{tij}. 4) By using normalized absolute error, we evaluated how closely the outputs of proposed and baseline methods, M_{tij}, were to the true values, M_{tij}^*.

Since the tendency of human flows (i.e. transition probability between cells) is expected to change with time of day,

---

[2] We use the car data collected by the smartphone car navigation application of NAVITIME JAPAN Co., Ltd. (http://corporate.navitime.co.jp/en/index.html)

we split the population data into 5 subsets based on time-of-day (6:00–9:00, 9:00–12:00, 12:00–15:00, 15:00–18:00, and 18:00–21:00). We then applied our models (and the baseline) to each subset. As a result, the model parameters including transition probabilities are common in each subset. It is reasonable that accuracy will be improved by choosing the time periods to suit human movement patterns. For example, an approach to infer the time periods of each subset that are suitable for given data was detailed in [Iwata *et al.*, 2017], and the same method can be applied to our proposal. However, our experiment focuses on "accuracy degradation due to high degree of freedom" and "the computational complexity problem". To concentrate on the key advances, we did not conduct an experiment on the subject of choosing optimum time subsets, but left it as future work.

5.2 Accuracy of Estimation

We compared proposed methods with the baseline method, which was described in Section 4.1. Note that the baseline method directly infers the elements of θ. For the baseline method, we set Γ_i to be a set of cells within L_∞-distance $K (K = 1, 5, 10)$ from cell i. For proposed methods, we set Γ_i to be a set of cells within L_1-distance 10 from cell i.

Results are shown in Figure 2. We first address the performance improvements attained by our two methods in total score (6:00-21:00). Of particular interest, our method is noticeably better in the time zone 6:00–9:00, in which there are many movements to distant cells due to commuting etc. Second, the superiority of our proposed methods over the baseline with $K = 1$ on the 5km × 5km grid space was stronger than that on the 10km × 10km grid space (23.5–26.3% and 6.5–12.7% rel. improvements on 5km × 5km and 10km × 10km, respectively). This thought to be because the finer the grid becomes, the more often the movements are to remote cells. Third, although estimation performance of the approximation method is slightly worse than that of exact method with the 10km × 10km grid space, they achieve almost the same performance with the 5km × 5km grid space. This indicates that the approximation method is a valid alternative to the exact method. We conducted experiments and verified that the proposed methods outperformed baseline methods using L_1-distance, though we omitted this result to enhance the clarity of the bar graphs. We found similar results using other evaluation metrics, such as MAPE (mean absolute percentage error).

5.3 Computation Time

We evaluated the computation time of the two proposed methods and the baseline method in the 5km×5km grid space. For all methods, we set Γ_i to be a set of cells within L_1-distance 10 from cell i. Figure 3 shows the relationship between computation time and NAE of each method in the time zones of 6:00–9:00 and 18:00–21:00. For each method, we performed estimation 10 times and obtained the average value. It shows that the approximate method can output accurate values more quickly than the exact method and the baseline method. For example, in the time zone 18:00–21:00, it took the exact method about 178 seconds to reach 0.55 for NAE. The approximate method, on the other hand, took only 38 seconds
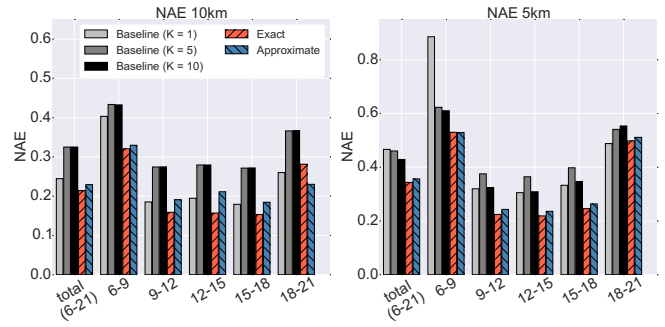

Figure 2: Results of experiments on estimation accuracy. We used NAE as evaluation metric for evaluating performance using two datasets with different grid size, 10km (left) and 5km (right).
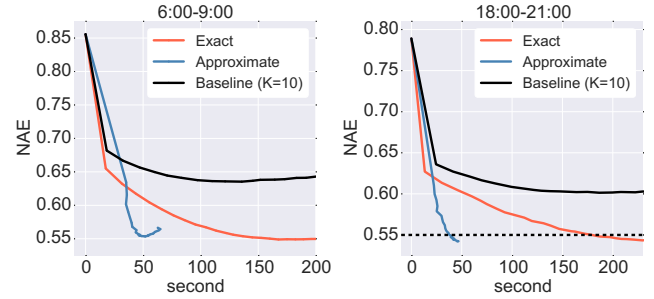

Figure 3: Results of experiments on computation time.

or so, which is about 4.7 times faster. Similar results are seen in all other time zones.

6 Conclusion

This paper proposed a new probabilistic model to estimate the movement of people from spatiotemporal population data. In order to consider movements to remote cells, there are two technical challenges: (i) poor estimation accuracy due to excessively high degrees of freedom of the model, (ii) large computation cost. We address these difficulties by using three factors to model transition probabilities. This enables us to reduce the degrees of freedom of the model and derive efficient estimation algorithms. Experiments confirmed the high estimation accuracy of our methods and light computation cost of the proposed approximate estimation algorithm.

References

[Acs and Castelluccia, 2014] Gergely Acs and Claude Castelluccia. A case study: Privacy preserving release of spatio-temporal density in paris. In *KDD*, pages 1679–1688, 2014.

[Boyd and Vandenberghe, 2004] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[Byrd *et al.*, 1995] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[Du *et al.*, 2014] Jiali Du, Akshat Kumar, and Pradeep Varakantham. On Understanding Diffusion Dynamics of Patrons at a Theme Park. In *AAMAS*, pages 1501–1502, 2014.

[Hess *et al.*, 2004] Ronald L. Hess, Ronald S. Rubin, and Lawrence A. West. Geographic Information Systems as a Marketing Information System Technology. *Decision Support Systems*, 38(2):197–212, 2004.

[Hoang *et al.*, 2016] Minh X. Hoang, Yu Zheng, and Ambuj K. Singh. FCCF: Forecasting Citywide Crowd Flows Based on Big Data. In *SIGSPATIAL*, pages 1–10, 2016.

[Hunter, 2003] David R. Hunter. MM algorithms for generalized Bradley-Terry models. *The Annals of Statistics*, 32(1):384–406, 2003.

[Iwata *et al.*, 2017] Tomoharu Iwata, Hitoshi Shimizu, Futoshi Naya, and Naonori Ueda. Estimating People Flow from Spatiotemporal Population Data via Collective Graphical Mixture Models. *ACM Transactions on Spatial Algorithms and Systems*, 3(1):1–18, 2017.

[Kiefer, 1953] Jack Kiefer. Sequential Minimax Search for a Maximum. *Proceedings of the American Mathematical Society*, 4(3):502–506, 1953.

[Kumar *et al.*, 2015] Ravi Kumar, Andrew Tomkins, Sergei Vassilvitskii, and Erik Vee. Inverting a Steady-State. In *WSDM*, pages 359–368, 2015.

[Luce, 1959] R. Duncan Luce. *Individual Choice Behavior: A Theorerical Analysis*. Jhon Wiley & Sons, Inc., New York, 1959.

[Maystre and Grossglauser, 2017] Lucas Maystre and Matthias Grossglauser. ChoiceRank: Identifying Preferences from Node Traffic in Networks. In *ICML*, pages 2354–2362, 2017.

[Sheldon and Dietterich, 2011] Daniel R. Sheldon and Thomas G. Dietterich. Collective Graphical Models. In *NIPS*, pages 1161–1169, 2011.

[Sheldon *et al.*, 2013] Daniel Sheldon, Tao Sun, Akshat Kumar, and Tom Dietterich. Approximate Inference in Collective Graphical Models. In *ICML*, pages 1004–1012, 2013.

[Sudo *et al.*, 2016] Akihito Sudo, Takehiro Kashiyama, Takahiro Yabe, Hiroshi Kanasugi, Xuan Song, Tomoyuki Higuchi, Shin'ya Nakano, Masaya Saito, and Yoshihide Sekimoto. Particle Filter for Real-time Human Mobility Prediction Following Unprecedented Disaster. In *SIGSPATIAL*, pages 1–10, New York, USA, 2016.

[Sun *et al.*, 2015] Tao Sun, Daniel Sheldon, and Akshat Kumar. Message Passing for Collective Graphical Models. In *ICML*, pages 853–861, 2015.

[Terada *et al.*, 2013] Masayuki Terada, Tomohiro Nagata, and Motonari Kobayashi. Population Estimation Technology for Mobile Spatial Statistics. *NTT DOCOMO Technical Journal*, 14(3):10–15, 2013.

[Witayangkurn *et al.*, 2013] Apichon Witayangkurn, Teerayut Horanont, and Ryosuke Shibasaki. The Design of Large Scale Data Management for Spatial Analysis on Mobile Phone Dataset. *Asian Journal of Geoinformatics*, 13(3), 2013.

[Xu *et al.*, 2017] Fengli Xu, Zhen Tu, Yong Li, Pengyu Zhang, Xiaoming Fu, and Depeng Jin. Trajectory Recovery From Ash: User Privacy Is NOT Preserved in Aggregated Mobility Data. In *WWW*, pages 1241–1250, 2017.

[Zhang *et al.*, 2017] Junbo Zhang, Yu Zheng, and Dekang Qi. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. In *AAAI*, pages 1655–1661, 2017.