ABSTRACT
Crowd movement guidance has been a fascinating problem in various fields, such as easing traffic congestion in unusual events and evacuating people from an emergency-affected area. To grab the reins of crowds, there has been considerable demand for a decision support system that can answer a typical question: "what will be the outcomes of each of the possible options in the current situation?". In this paper, we consider the problem of estimating the effects of crowd movement guidance from past data. To cope with limited amount of available data biased by past decision-makers, we leverage two recent techniques in deep representation learning for spatial data analysis and causal inference. We use a spatial convolutional operator to extract effective spatial features of crowds from a small amount of data and use balanced representation learning based on the integral probability metrics to mitigate the selection bias and missing counterfactual outcomes. To evaluate the performance on estimating the treatment effects of possible guidance, we use a multi-agent simulator to generate realistic data on evacuation scenarios in a crowded theater, since there are no available datasets recording outcomes of all possible crowd movement guidance. The results of three experiments demonstrate that our proposed method reduces the estimation error by at most 56% from state-of-the-art methods.

KEYWORDS
Deep Learning, Causal Inference, Multi-Agent Simulator
We use an uppercase letter to denote a random variable (i.e., $X$), which takes advantage of two recent developments in deep representation learning in spatial data analysis and causal inference. Because the locational distribution of crowds has a significant impact on the effectiveness of movement guidance, SC-CFR aggregates crowd distribution information by means of the spatial convolutional layer [19, 41] and predicts the outcomes of movement guidance with high accuracy even with a limited number of data. On the other hand, to cancel the data biases made by past decision-makers, we resort to the causal inference framework that has been successfully applied to answer the what-if problems in various domains, such as healthcare [24], economics [13], and education [42]. In particular, SC-CFR employs balanced representation learning based on integral probability metrics, which mitigates selection bias and missing counterfactual outcomes [24].

Evaluating the performance of estimating the possible guidance’s treatment effects requires datasets recording outcomes of all possible crowd movement guidance. Since there are no datasets available for our purpose, we used a multi-agent simulator [36] to generate realistic data on evacuation scenarios in a crowded theater. From these datasets, we conducted experiments for estimating the treatment effects on outcomes, such as the maximum, average, and standard deviation of evacuation times required by crowds. We demonstrate that our proposed method reduced the estimation error by at most 56% from state-of-the-art baseline methods that do not consider spatial attributes.

Our contributions are summarized as follows:

- We first address the what-if problem in crowd movement guidance as causal effect estimation from a relatively small amount of biased data.
- We propose SC-CFR, which takes advantage of two recent developments in deep representation learning in spatial data analysis and causal inference.
- We demonstrate that our method reduces the estimation error by at most 56% from state-of-the-art baselines by experiments with a multi-agent simulator.

## 2 PRELIMINARIES

We use an uppercase letter to denote a random variable (i.e., $X$), a lowercase letter to indicate a realization $x$ drawn from a particular distribution (i.e., $x \sim P(X)$), and a calligraphy letter $X$ to represent a sample space (i.e., $x \in X$).

### 2.1 Crowd Movement Guidance

Let us consider a place or area with crowds of freely moving people. When the place is crowded, a manager should want to carry out a crowd guidance plan to reduce the congestion caused by crowds moving toward other places. For example, if a park hosts a big festival with a sell-out crowd, a park manager has to make a decision on guidance to avoid traffic jams. Another example is of a manager working in a stadium who needs to make a decision on guidance to safely and quickly evacuate crowds during a fire emergency. A crowd guidance plan can comprise various types of actions, such as route recommendations for crowds or a modification of road width, which could have complicated interactions.

Assume that a situation of crowds, such as their current locations, is observed as covariates. We denote a situation of crowds (covariates) as $X \in \mathbb{R}^d$, where $d$ is the number of people. A guide (treatment) is $Z \in \mathcal{Z} \subset \{0, 1\}^t$ that consists of $t$ possible actions, where $\mathcal{Z}$ is a set of possible guides. A guide can select single or multiple actions (i.e., a combination of actions); therefore, $|\mathcal{Z}|$ grows exponentially as $t$ increases. We denote a potential outcome as $Y \in \mathbb{R}$, such as an average moving cost or the maximum movement cost consumed by crowds. For example, let us suppose an evacuation scenario from a place. Given an initial situation of crowds $x \sim P(X)$, a decision-maker, who needs to guide crowds to a safe place, selects a treatment $z \sim P(Z|X)$, after which crowds start to move. We can observe an outcome $y \sim P(Y|X,Z)$, such as evacuation times, after finishing the scenario. This procedure is built upon by the Rubin-Neyman potential outcomes framework [22]. We show a framework for crowd movement guidance in Figure 2.

### 2.2 Causal Effect Estimation

Suppose that there is a healthcare record (covariate) of a patient, and a doctor needs to choose whether to prescribe medicine, which we will call a treatment. We can observe a future health condition of the patient as a subsequent outcome. Causal effect estimation is a task for estimating outcomes given not actual treatments and to assess the difference in outcomes between treatments. In this paper, we consider the situation of crowds as covariates, guidance as a treatment, and costs of crowd movements as outcomes. Effects of a guide should differ from situation to situation since it depends on the crowd conditions and types of events. Thus, decision-making on guide planning is one of the most difficult tasks for managers, and they need a useful tool to help their decision-making. Surprisingly, the effect of guidance has not been considered in the existing literature.

We suppose that two arbitrary guide plans $A$ and $B$ correspond to guides $z$ and $z'$, respectively. Given covariates $x$ and a guide $z$, we denote the cost consumed by crowds as $Y_x(z)$. The conditional average treatment effect (CATE) for covariates $x$ can be defined as $\mathbb{E}[Y_x(z)] - \mathbb{E}[Y_x(z')]$, where we use $\mathbb{E}$ to denote the expectation over $Y$. Intuitively, this quantity indicates how much of an advantage (or disadvantage) plan $A$ has in guiding the crowd movements compared with plan $B$. Knowing this quantity helps a decision-maker compare the difference between treatments for a particular situation $x$; however, we cannot directly calculate CATE from observed outcomes because we can only observe the outcome of one of the treatments.
2.3 Selection Bias and Missing Counterfactuals

Estimating the outcome of guidance requires a deep understanding of the complex relationships within covariates and guidance. One of the difficulties in estimating treatment effects on crowd movement is that an observed set of guidance is biased in the given dataset, because the manager usually selects guidance according to their policy. Thus, even if the observed covariates are random, we cannot observe the outcome with a pair of covariates and guidance that managers do not prefer. Another obstacle is that we can only observe a factual outcome, but cannot observe counterfactual outcomes. Usually, we can only deploy a single guide in a situation, and thus the outcomes for the other possible guides remain missing.

Formally, in an observed study, we can obtain only one factual outcome $y$ depending on a selected treatment $z$, and thus never see counterfactual outcomes $y^{ft}$ corresponding to the other treatments that are not selected by the decision-maker. More formally, when a decision-maker chooses a treatment $z$, a factual outcome $y$ is equal to the outcome $y_x(z)$, and the counterfactual outcomes $y^{ft}$ are $y_x(z^*)$, $z^* \in Z \setminus z$. We suppose that a decision-maker has a policy for choosing a treatment $z$ based on a situation $x$, and thus, observed treatments are not at random and always biased. This bias, which is called a selection bias in the Rubin-Neyman framework, leads to a simple model for estimating outcomes, not accurate and biased results. Thus, we need to develop a method that learns a model for predicting outcomes and can simultaneously cancel the effects of the selection bias.

In this paper, we make the common assumption called strong ignorability in the Rubin-Neyman framework.

**Assumption 1 (Stable Unit Treatment Value Assumption).** The potential outcomes for any situations do not vary with the treatment assigned to other situations, and, for each situation, there are no different forms or versions of each treatment level, which lead to different potential outcomes.

This assumption emphasizes the independence of each situation and that there are no interactions between situations.

**Assumption 2 (Ignorability).** Given the covariate, $X$, treatment assignment $Z$ is independent to the potential outcomes, i.e., $Z \perp Y(x) \mid X$, $\forall z \in Z$.

The ignorability assumption is also referred to the unconfoundedness assumption. With this unconfoundedness assumption, the situations with the same covariates $X$, their treatment assignment can be viewed as random.

**Assumption 3 (Positivity).** For any value of $X$, treatment assignment is not deterministic: $P(Z = z \mid X = x) > 0$, $\forall z$ and $x$.

This assumption implies that all the factors determining the outcome of each treatment are observed. From the above assumptions, we follow strong ignorability assumption that is referred to the no-hidden confounders assumption. We formalize this assumption by using the standard strong ignorability condition:

**Assumption 4 (Strong Ignorability).** $Z \perp Y(x) \mid X$, $\forall z \in \{0, 1\}$ and $P(Z = z \mid X = x) > 0$, $\forall z$ and $x$.

The strong ignorability assumption is a sufficient condition for CATE function $f$ to be identifiable [13].
takes the inner product or measures the cross-correlation between an input array \( a \) and a kernel \( k \in \mathbb{R}^{k_1 \times k_2} \), defined as

\[
(k \ast a)(i, j) = \sum_{m=1}^{k_1} \sum_{n=1}^{k_2} a_{i+m, j+n} k_{m,n}.
\]

(1)

The average pooling operator is the convolution operator whose kernel values are set to 1. We employ two repetitions of a convolution operator and an average pooling operator as \( g(x) \) in this paper (See Figure 3).

### 3.3 Balanced Representation

If we do nothing to address the observation biases in data, the distribution of the extracted spatial feature \( \Phi_x \) is biased depending on \( z \), which could make the prediction accuracy of the model deteriorate. To mitigate the bias and make \( \Phi_x \) a balanced representation, we utilize the integral probability metrics (IPM) [20, 27] in this paper. Let us consider a binary treatment case \( \mathcal{Z} = \{0, 1\} \): IPM can measure distance between two distributions. Given two probability density functions \( P(\Phi_x|z) \) and \( P(\Phi_x|\bar{z}) \), for \( \mathcal{S} \subseteq \mathbb{R}^d \) and \( G \), a family of functions from \( \mathcal{S} \) to \( \mathbb{R} \), IPM is defined as

\[
\text{IPM}_G(p, q) = \sup_{g \in G} \left| \int_{\mathcal{S}} g(x) (p(x) - q(x)) dx \right|.
\]

(2)

Intuitively, minimizing IPM makes the distributions of the extracted spatial features for the factual and counterfactual treatments hard to distinguish, and thus the distribution of the training data becomes close to those of RCTs [10, 24].

In this paper, we consider a case where a guiding plan \( z \in \mathcal{Z} = \{0, 1\}^d \) consists of multiple actions, such as route recommendations for crowds and a modification of the road width. However, the IPM compares only two distributions, and therefore, is not directly applicable to multiple treatments [39]. One of the most natural extensions of IPM to multiple treatments is to measure IPM between all possible pairs of treatments; however, this requires \(|\mathcal{Z}|^2\) IPM terms and is not tractable for a large number of treatments. To cope with this problem, we utilize another simple extension of IPM that considers the most frequent treatment in a mini-batch as the control treatment and only use the IPMs between the control and the others.

\[
\sum_{z' \in \mathcal{Z} \setminus z} \text{IPM}_G(p(\Phi_x|z), q(\Phi_x|z')).
\]

(3)

### 3.4 Outcome Prediction

Finally, we utilize the spatial representations balanced for all the treatments to construct a predictive model \( h(\Phi_x, z) \) to infer an outcome \( y \). The original counterfactual regression [24] and its variants [23] employ multi-head networks where each multi-layer network corresponds to a treatment \( z_i \) and shares common base layers to avoid the risk of losing treatment information. However, since the total number of treatments \( |\mathcal{Z}| \) rapidly grows with the number of actions, and training all of the heads requires a considerable amount of observations that are not available in our setting, we employ a single-head network instead built on a multi-layer perceptron (MLP) that takes a concatenation of \( \Phi_x \) and \( z \) as the input.

\[
h(\Phi_x, z) = \text{MLP}(\Phi_x, z).
\]

(4)

The single-head network architecture has showed better performance when the number of treatments is more than a dozen [30].

To train the SC-CFR model, we find the optimal parameter that minimizes the loss function.

\[
\sum_{i=1}^{N} \|y_n - \text{MLP}(\Phi_{x_n}, z_n)\|^2 + \lambda \sum_{z' \in \mathcal{Z} \setminus z} \text{IPM}_G(p(\Phi_x|z_n), q(\Phi_x|z'_n)).
\]

(5)

where \( \lambda > 0 \) is a hyper parameter for controlling the effect of IPM. We employ the ADAM optimiser to optimize parameters in a mini-batch manner [16]. Among several candidates for the IPM, we employ the maximum mean discrepancy (MMD) [9, 27] for its simplicity in this paper.

### 4 RELATED WORK

Modeling and simulating large-scale crowd movements [3], including traffic network flows in cities [25, 33] and at sea [26], and evacuation from buildings [7, 17, 32, 43] have been actively studied in the field of multi-agent systems. Researchers have exploited multi-agent simulators to find optimal decision-making policies for crowd management. Comparisons of the properties of various types of simulators are found in [21]. Most of the existing research focuses on bottom-up decision-making by individual crowds, i.e., each agent making decisions based on individual or partial information, to achieve one’s goals. This study focuses on the indirect control of crowd movement through top-down decision-making by managers and the estimation of its effects.

A convolutional operator is a feature extraction method developed in computer vision for capturing low and high level image features [8]. Researchers in spatial data analysis have recently applied convolutional operators to handle the spatial correlations in urban areas [19]. They have extracted local and city-wide global features by stacking spatial convolutional operators and utilized it to solve their domain-specific problems [18]. With such rich features, many researchers have reported significant improvements in the predictive accuracy on forecasting future amounts and speeds of transportation networks [40, 41] and rides of on-demand ride services [15], to name a few. Based on their results, we utilize a spatial convolutional operator for extracting effective local spatial features of crowds from a small amount of data.

In the field of causal inference, novel methods for estimating the causal effect on individual and population levels have been proposed [12, 37]. The bayesian additive regression tree (BART) [11] is one of the most popular methods built on a simple decision tree model [2, 34] and it shows state-of-the-art performance. Recently, in the field of machine learning, novel methods based on deep neural networks to remove biases in data introduced by past decision-makers have been proposed [1, 5, 14, 24, 38]; they use representation learning techniques to learn balanced representations of confounders. In particular, the counterfactual regression (CFR) [24] and its various extensions [10, 23, 39] have received widespread interest due to its significant performance improvement in estimating treatment effects compared to existing methods, such as BART. However, most of the existing studies have not considered the use of spatial information to address data deficiencies and observation biases. In this paper, we use spatial convolution to extract effective spatial features for estimating treatment effects from the
distribution of crowds given as images [19, 41]. The problem of estimating treatment effects is closely related to bandit feedback [4, 28] and off-policy reinforce learning [29]. Some of them assume that the policies of the agents used to collect data are known [10, 12]. Meanwhile, we do not assume prior information about the data collection mechanisms.

5 EXPERIMENTS

5.1 Experimental Setup

Evaluating the performance of estimating possible guidance treatment effects requires datasets recording the outcomes of all possible crowd movement guidance. Since such datasets are not available, we used an open-source multi-agent simulator [36] to generate realistic data on evacuation situations in a crowded theater. We utilized the actual structure of an opera house, namely New National Theatre, Tokyo in our simulator, whose seating capacity is \( d = 868 \). We show the building structures in Figure 7, in which the yellow blocks correspond to the seats. The blue lines and red texts represent actions of treatments explained, as in the following paragraph. Crowds evacuate from the auditorium to the outside using one of the six exits. We conducted a real evacuation drill at the opera house and measured the evacuation movements of 570 people using eight stereo cameras. Then, we employed the simulator [36] to generate crowd movement. We used the default setting parameters and set agents at the same initial locations as in the drill. We confirmed that the absolute error of the maximum evacuation time between measured and simulated data was less than five percent. We show images of an evacuation drill held in the opera house in Figures 4, 5, and 6.

In our experiments, we considered two types of actions. The first type of action was route guidance to guide the audience to the evacuation exits. If no route guide is placed, each audience evacuates from the nearest exit. We employed a route guide whose maximum evacuation time was the smallest than other guides when all seats were occupied and all doors were open. The second type of the actions was door operations that determine whether the door of an exit to be fully open or half-open. In our scenario, we considered a risk, such as spreading fires, and allowed only two of six doors to be fully open. We show snapshots of simulations having the same covariates but with and without route guidance in Figures 9 and 8, where the green and red dots represent agents that are moving and stopping, respectively. We can observe the positive effect of route guidance by comparing the snapshots at 160 sec; without route guidance, congestion occurred (Figure 8), while it is dissolved if we give route guidance (Figure 9). Our codes and datasets are publicly available at a repository.

5.2 Data Generations

We generated crowd movement datasets using our simulator with the following settings. We first split the auditorium into four blocks (Blocks A, B, C, and D in Figure 7) and set the occupancy rate of each block from \{0.1, 0.5, 0.9\}. We randomly set the occupation of each seat with the occupancy rate and constructed covariates \( x \) by setting \( x_i \) to 1 if the \( i \)-th seat was occupied and set to 0 otherwise. We ran the seat selection ten times with different random seeds. Then, we put agents on the occupied seats and ran evacuation simulations for all possible treatments \( z \in \mathcal{Z} \), where \( |\mathcal{Z}| = 2 \cdot (6 \cdot 5)/2 = 30 \). We omitted simulations where the total number of people was less than 400, since the guidance was not effective in such situations. The total number of data \( N \) was 423 and each data has 30 possible treatments and their corresponding outcomes. We observed three outcomes for each run: the maximum, average, and standard deviation of evacuation times required by each person. Then, we added observation noises to the outcomes that were randomly sampled from a normal distribution \( \mathcal{N}(0, \sigma^2) \) where \( \sigma = 2 \). We show the distributions of the outcomes in Figures 10, 11, and 12. The red and blue lines correspond to the outcomes with and without route guidance, respectively.

To select a factual treatment \( z \) for guidance, we employed two independent probabilistic policies (the propensity scores). We set the first dimension of treatment \( z \in \{0, 1\}^2 \) to represent the action
We randomly sampled whether to use the route guide using a distribution \( p(z_{\text{guide}} = 1) = 1/(1 + \exp(-\sum_{i=1}^d x_i/d + 1)) \). We set \( z_{\text{guide}} \) to 1 when the route guidance is given. To decide which exit doors to be fully open, we calculated the populations of occupied seats within 8 meters \( \sum_{i \in D_j} x_i / |D_j| \) from each door, where \( D_j \) is the number of the seats within 8 meters from the \( j \)-th door. Finally, we randomly sampled two doors to be open by using a multinomial distribution whose parameters were set to those populations. We treated all the other treatments as counterfactual treatments.

### 5.3 Model Settings

We randomly sampled 90% of the seat assignments as a training dataset and used the rest as a test dataset. As the baseline methods, we employed nine supervised learning methods [6]: Lasso [31], the ridge regression (Ridge), the support vector regression (SVR), the random forest (RF), the gradient boosting regressor (GBR), the bayesian additive regression tree (BART) [11], and the multi-layer perceptron (MLP) consisting of five layers. Since these methods were not applicable to identify differences between covariates and all possible treatments. We define \( \Phi \) as the goal is to estimate the outcomes of all possible treatments. The treatment-agnostic representation network (TARNET) and the counterfactual regression (CFR) are recently proposed neural network-based causal effect estimation methods in machine learning fields [24]. For TARNET and CFR, we used a two-layer MLP to learn the representation of \( x \) and a three-layer MLP whose inputs were a concatenation of the extracted representation and treatments to predict the outcome. We employed two variants of our proposed methods. SC-CFR is our proposed method using IPM, and a spatial convolutional treatment-agnostic representation network (SC-TARNET) is the proposed method without IPM (\( \lambda = 0 \)). We constructed two-dimensional arrays \( x \in \mathbb{R}^{22 \times 42} \) from covariates based on seat layouts. We show examples of the arrays in Figure 13. For our methods, we utilized two repetitions of a block that consisted of a convolutional operator and an average pooling operator, whose kernel size \( k \) and paddings were set to 3 to extract the spatial feature \( \Psi_x \). Then, we used a three-layer MLP to predict the outcome. We selected model architectures and hyperparameters by minimizing the training errors.

### 5.4 Metrics

To compare the performance of models, we employed several metrics that could measure the accuracy of predicted outcomes \( \hat{y} \) under the potential outcome framework. We evaluated models in two different settings; the within-sample setting, where the task is to estimate outcomes for all scenarios in the training dataset, evaluates what happens if a decision-maker chose other treatments in the past evacuation events. On the other hand, the out-of-sample setting, where the task is to estimate outcomes for scenarios in the test dataset, corresponds to the case of a new evacuation event, and the goal is to estimate the outcomes of all possible treatments.

- **Root mean squared error (RMSE).** RMSE is a widely used metric in supervised learning problems. We employed RMSE to measure the accuracy of each predicted outcome depending on the covariates and all possible treatments. We define \( e_{\text{RMSE}} \) as

\[
\frac{1}{|Z|} \sum_{z \in Z} \sum_{n=1}^N \left( \hat{y}_{x_n}(z_i) - \mathbb{E}[Y_{x_n}(z)] \right)^2)^{1/2},
\]

where \( Z^\prime \) is a set of whole pairs of possible guiding plans.

- **Precision in estimation of heterogeneous effect (PEHE).** In the binary setting, PEHE measures the ability of a predictive model to estimate the difference between true and predicted outcomes between two treatments \( z_0 \) and \( z_1 \) for covariates \( x \). We define PEHE as the RMSE between the true difference of the outcomes of two treatments \( z_1 \) and \( z_j \), that is, \( \mathbb{E}[Y_{x_n}(z_1)] - \mathbb{E}[Y_{x_n}(z_j)] \), and the predicted difference, \( \hat{y}_{x_n}(z_i) - \hat{y}_{x_n}(z_j) \). Obtaining a better PEHE requires accurate estimations of both the factual and counterfactual outcomes. We set \( e_{\text{PEHE}} \) as

\[
\frac{1}{N} \sum_{n=1}^N \left( \hat{y}_{x_n}(z_1) - \hat{y}_{x_n}(z_j) - (\mathbb{E}[Y_{x_n}(z_1)] - \mathbb{E}[Y_{x_n}(z_j)]) \right)^2)^{1/2}.
\]
We define $\epsilon$.

ATE is not as important as PEHE for models optimised for CATE estimation, but it can be a useful indicator of how well an estimator performs at comparing two treatments across the entire population. Estimation results of $\epsilon_{\text{mATE}}$ in Tables 1 and 3, while BART showed the best performance in Table 1.

From Tables 1, 2, and 3, we confirmed that our proposed SC-CFR achieved the best performance on $\epsilon_{\text{RMSE}}$ and $\epsilon_{\text{mPEHE}}$. SC-TARNET came in the second place. GBR or BART placed third. In Table 1, SC-CFR reduced the estimation error of $\epsilon_{\text{mPEHE}}$ out-of-sample by 33% and 38% from GBR, and 55% and 56% from Ridge, respectively.

The improvements of SC-CFR on $\epsilon_{\text{RMSE}}$ and $\epsilon_{\text{mPEHE}}$ for both within-sample and out-of-sample settings indicated that our method successfully extracted spatial features and learned balanced representations, thus providing more accurate predictions of outcomes than the state-of-the-art baselines. Our methods achieved the best estimation results of $\epsilon_{\text{mATE}}$ in Tables 1 and 3, while BART showed the best performance in Table 2. Since the distribution of the standard deviations had a relatively simpler shape than the others, BART, which is a simpler non-linear method than ours, was sufficient to estimate ATE.

By comparing MLP and TARNET, TARNET showed better scores than MLP because of its representation learning mechanism, which MLP does not have. The performance of SC-CFR was significantly better than that of CFR. With these results, we could argue that the spatial convolutional operators worked correctly and greatly contributed to this improvement. As regarding the effect of bias cancellation, we observed that IPM contributed to the performance improvements of CFR from TARNET and SC-CFR from SC-TARNET. This result indicates that a balanced representation was effective in precisely estimating CATE of crowd movement guidance, which is consistent with existing literature [24].

**Average treatment Effect (ATE).** We employed the error on estimating a conditional ATE. In contrast to PEHE, ATE considers only the average difference of treatment effects over the population. ATE is not as important as PEHE for models optimised for CATE estimation, but it can be a useful indicator of how well an estimator performs at comparing two treatments across the entire population. We define $\epsilon_{\text{ATE}}$ as

$$\frac{1}{N} \sum_{n=1}^{N} (\hat{y}_{x_n}(z_i) - \hat{y}_{x_n}(z_j)) - \frac{1}{N} \sum_{n=1}^{N} (E[Y_{x_n}(z_i)] - E[Y_{x_n}(z_j)]).$$

**Multiple Treatments.** Let us consider a general case where combinations of multiple actions are available for making a guiding plan. By following [23], we extended PEHE and ATE to be applicable to multiple treatments by considering the average PEHE and ATE between every possible pair of guiding plans defined as

$$\epsilon_{\text{mPEHE}} = \frac{1}{\binom{2}{2}} \sum_{(z_i, z_j) \in Z} \epsilon_{\text{PEHE}}, \quad \epsilon_{\text{mATE}} = \frac{1}{\binom{2}{2}} \sum_{(z_i, z_j) \in Z} \epsilon_{\text{ATE}}.$$

### 5.5 Results

Tables 1, 2, and 3 show the results of prediction of the maximum, average, and standard deviation of evacuation times, respectively. Each column represents the average and the standard deviation of each metric. A bold letter indicates a method that achieved the smallest (best) error in each column. To assess the effect on bias cancellation, we employed the t-test on two related samples of the scores between TARNET and CFR, and SC-TARNET and SC-CFR. The values with a single star or double stars indicate that the p-values between the two models were $p < 0.10$ and $p < 0.05$, respectively.

### 6 CONCLUSION

In this paper, we focused on the problem of estimating the effects of crowd movement guidance from a relatively small amount of biased data to answer a typical question posed by decision-makers: "what will be the outcomes of each of the possible options in the current situation?". We proposed a Spatial Convolutional Counterfactual Regression (SC-CFR), which takes advantage of two recent developments in deep representation learning in spatial data analysis and causal inference. We utilized spatial convolutional operators to extract effective spatial features of crowds from a small amount of data, and applied balanced representation learning using IPM to mitigate the selection bias and missing counterfactual outcomes. With
datasets generated by a multi-agent simulator on evacuation scenarios in a crowded theater, we conducted experiments to estimate treatment effects on three types of outcomes of evacuation times required by crowds. We demonstrated that SC-CFR reduced the estimation error by at most 56% from the state-of-the-art baseline methods that did not consider spatial attributes.

**Acknowledgement** This work was supported by JST, PRESTO Grant Number JPMJPR20C5, Japan and JSPS KAKENHI Grant Numbers 20H00609, Japan.

---

### Table 1: Results on estimating the maximum evacuation time.

| Method     | Within-sample | Out-of-sample |
|------------|---------------|---------------|
| Lasso      | 17.334 (0.282) | 18.524 (2.116) |
| Ridge      | 16.886 (0.576) | 24.369 (1.962) |
| SVR        | 14.895 (0.179) | 15.519 (0.714) |
| RF         | 13.595 (0.444) | 14.824 (0.779) |
| GBR        | 11.617 (0.503) | 12.841 (0.490) |
| BART       | 11.620 (1.299) | 12.508 (1.357) |
| MLP        | 17.911 (0.310) | 21.243 (1.628) |
| TARNET     | 12.992 (0.227) | 13.981 (0.863) |
| SC-TARNET  | 7.748 (0.746)  | 7.920 (0.595)  |
| SC-CFR     | 7.658 (0.787)  | 7.816 (0.579)  |

### Table 2: Results on estimating the average evacuation time.

| Method     | Within-sample | Out-of-sample |
|------------|---------------|---------------|
| Lasso      | 6.497 (0.167) | 6.889 (1.604) |
| Ridge      | 3.075 (0.093) | 5.024 (1.448) |
| RF         | 1.985 (0.103) | 3.467 (0.903) |
| GBR        | 2.336 (0.145) | 2.793 (2.091) |
| BART       | 2.298 (0.133) | 2.503 (0.215) |
| MLP        | 4.132 (0.078) | 5.688 (1.543) |
| TARNET     | 2.589 (0.077) | 3.030 (0.301) |
| SC-TARNET  | 1.849 (0.249) | 1.929 (0.252) |
| SC-CFR     | 1.922 (0.250) | 1.869 (0.237) |

### Table 3: Results on estimating the standard deviation of evacuation time.

| Method     | Within-sample | Out-of-sample |
|------------|---------------|---------------|
| Lasso      | 6.538 (0.130) | 6.699 (0.472) |
| Ridge      | 3.571 (0.109) | 5.487 (0.384) |
| SVR        | 3.455 (0.100) | 3.420 (0.139) |
| RF         | 3.412 (0.237) | 3.649 (0.250) |
| GBR        | 2.565 (0.164) | 2.832 (0.183) |
| BART       | 2.567 (0.095) | 2.772 (0.202) |
| MLP        | 4.427 (0.127) | 5.157 (0.314) |
| TARNET     | 2.961 (0.144) | 3.280 (0.150) |
| SC-TARNET  | 1.889 (0.159) | 1.933 (0.239) |
| SC-CFR     | 1.867 (0.141) | 1.954 (0.229) |
