scientific reports

OPEN

Chemical property prediction under experimental biases

Yang Liu & Hisashi Kashima

Predicting the chemical properties of compounds is crucial in discovering novel materials and drugs with specific desired characteristics. Recent significant advances in machine learning technologies have enabled automatic predictive modeling from past experimental data reported in the literature. However, these datasets are often biased because of various reasons, such as experimental plans and publication decisions, and the prediction models trained using such biased datasets often suffer from over-fitting to the biased distributions and perform poorly on subsequent uses. Hence, this study focused on mitigating bias in the experimental datasets. We adopted two techniques from causal inference combined with graph neural networks that can represent molecular structures. The experimental results in four possible bias scenarios indicated that the inverse propensity scoring-based method and the counter-factual regression-based method made solid improvements.

Predicting the chemical properties of compounds is crucial in discovering novel materials and drugs with specific desired characteristics. Various computational approaches, including those based on density functional theory, have been widely used to accelerate the discovery process; however, they remain expensive and time consuming. In recent decades, researchers have shifted to data-driven approaches by fast-progressing machine learning technologies[1]. Recently, this trend has been further accelerated by the remarkable development of deep learning; in particular, graph neural networks (GNNs)[2,3] have achieved remarkable performance in predicting chemical properties via automatic feature extraction from molecular structures represented as graphs[4–10]. Their applications have expanded to various tasks, such as molecular generation[11–13], molecular explanation[14,15], and analysis of inter-molecular interactions[16,17].

Accurate predictive models often rely on large-scale labeled datasets; they are frequently collections of knowledge (e.g., experimental results reported on scientific papers) that are the product of extensive scientific efforts. Unsurprisingly, scientists do not uniformly sample molecules from a large chemical space at random nor based on their natural distribution (i.e., the actual distribution of molecules existing in the chemical space) . Rather, their decisions on experimental plans or publication of results are biased due to physical, economic, or scientific reasons. For instance, a large proportion of molecules are not investigated experimentally because of molecular mechanics-related factors, such as solubility[18], weights[19], toxicity[20], and side effects, or molecular structure-related factors, such as crystals[21]. In the pharmaceutical domain, drug likeness is an important factor for target selection, as exemplified by "Lipinski's rule of five"[22]. Further, concerns about the cost and availability of molecules can be some reasons to exclude certain groups of molecules. Conversely, popularity considerations based on current research trends[23] and the experimental methods wherein each lab specializes[24] influence the selection of compounds. These propensities related to researchers' experience and knowledge can contribute to more efficient search and discovery in the chemical space; however, they influence the data in an undesirable manner. Biases from human scientific research result in datasets that are sampled from distributions that differ from the natural ones. Thus, prediction models trained using such biased datasets suffer from over-fitting to the biased distributions, leading to poor performance on subsequent uses [25–28].

Evidence on the existence of bias in scientific experiments and their harmful effects has been reported [18–28]. However, almost no attempts to address this challenge have been found. The problems of learning prediction models, when the distributions of the training and test datasets are different, are called domain adaptation, covariate shift adaptation[29], or transfer learning[30], and they have been a major topic in machine learning. Recently, deep neural networks for domain adaptation based on the concept of domain-invariant representation learning were proposed[31–37], which have been primarily applied in the study of images. The problem of biased observations is also discussed in the context of causal inference. Inverse propensity scoring (IPS) is a general method from causal inference[38,39], which has been successfully applied to various applications such as recommender systems[39,40], natural language processing[41], and treatment estimation in different fields ranging from healthcare[42], economy[43], and education[44]. Meanwhile, the domain-invariant representation learning concept is also introduced in the

Department of Intelligence Science and Technology, Kyoto University, Kyoto 606-8501, Japan. email: liuyang@ml.ist.i.kyoto-u.ac.jp

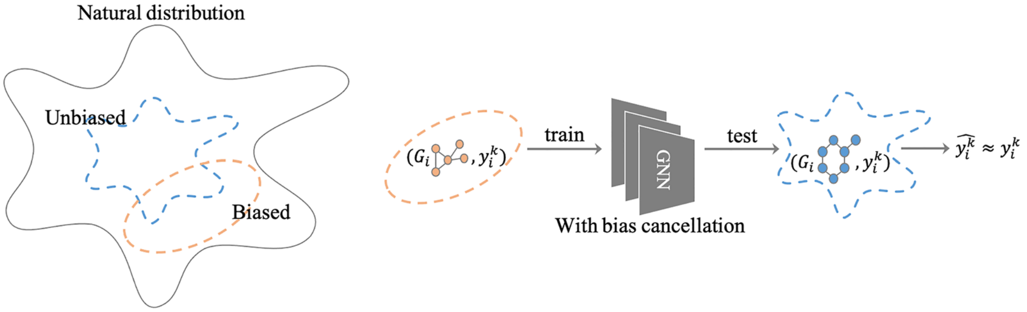

Figure 1. Main aim of our study. The left-hand figure shows the biased and unbiased distribution compared with the natural universe distribution of a chemical domain or sub-domain. The right-hand figure shows the proposed methods to train GNNs for chemical property prediction. Our aim was to apply bias cancelling techniques for GNNs to achieve significantly lower errors (i.e., MAE) when tested on a randomly sampled test dataset, whose distribution is similar to nature. G is the molecular graph, and y^k is the value of the k-th chemical property.

context of causal inference[45]. Counter factual regression (CFR) is a classic method which has been successfully applied for predicting individual treatment effect[45–47] by obtaining balanced representation such that the induced treated and control distributions look similar.

In this study, we focused on mitigating the sources of bias in experimental datasets by applying the IPS and CFR techniques from causal inference. With the IPS approach, we first estimate the propensity score function, which represents the probability of each molecule to be analysed, and then estimate the chemical property prediction model by weighting the objective function with the inverse of the propensity score. The CFR approach consists of one feature extractor, several treatment outcome predictors and one internal probability metric, where the feature extractor obtains features that aid the treatment outcome predictors and the internal probability metric, and the entire network is optimized in an end-to-end manner. Hassanpour et al.[47] further introduce an importance sampling weight estimator to improve the CFR architecture.

Both approaches were implemented over a GNN to study the molecular structures of the compounds. To the best of our knowledge, we are the first to combine chemical property prediction based on graph deep learning and sampling bias correction techniques. Figure 1 shows the main aim of this study.

Our experiments used two well-known large-scale dataset QM9[48] and ZINC[49], and two relatively smaller datasets ESOL and FreeSolv[50]. QM9 consists of exhaustively enumerated small-molecule structures associated with 12 fundamental chemical properties[51], In contrast, molecule structures in ZINC, ESOL, and FreeSolv are far less than exhaustively enumerated and they are associated with one property for each dataset. Because determining how a publicly available dataset is truly affected by bias is impossible, we simulated four practical biased sampling scenarios from the dataset, which introduced significant biases in the observed molecules. Under each biased sampling scenario, we validated our proposed models in predicting 15 chemical properties using 15 regression problems. The experimental results indicated that both the two-step IPS approach and the more modern end-to-end CFR approach improved the predictive performance in all the scenarios on most of the targets with statistical significance compared with the baseline method , and in addition, the CFR approach outperformed the IPS approach on most of the targets.

In summary, the main contributions of this paper are as follows:

- We first address the problem of predicting properties of chemical compounds under experimental biases.
- We introduce two bias mitigation techniques, IPS and CFR, combined with GNN-based prediction of chemical properties.
- We validated the two proposed approaches using various experimentally biased sampling scenarios and demonstrated that both of them improves the predictive performance significantly.

Results

Problem setting of predicting chemical properties under experimental biases.

We assume a training dataset $\mathscr{D}^{\text{train}} = \{(G_i, y_i)\}_{i=1}^{N}$ that includes N molecular graphs, where $G_i \in \mathscr{G}$ is a molecular graph (biasedly) sampled from a large chemical space $\mathscr{G}$, and $y_i \in \mathbb{R}$ is the target chemical property. Each molecular graph $G_i = (\mathscr{V}_i, \mathscr{E}_i, \sigma_i) \in \mathscr{G}$ has a set of nodes (i.e., atoms) $\mathscr{V}_i$, a set of edges (i.e., bonds) $\mathscr{E}_i \subseteq \mathscr{V}_i \times \mathscr{V}_i$, and a label function $\sigma : \mathscr{V}_i \cup \mathscr{E}_i \to \mathscr{L}$. Here, $\mathscr{L}$ is the set of possible node labels (i.e., atom types such as hydrogen and oxygen) and edge labels (i.e., bond types such as double bond and aromatic bond). Our aim is to obtain a prediction function $f : \mathscr{G} \to \mathbb{R}$ that predicts a particular target chemical property over the chemical space $\mathscr{G}$ that we are interested in. We assume that we have a uniformly randomly sampled part of (or possibly the entire) $\mathscr{G}$, consisting of M molecules $\mathscr{D}^{\text{test}} = \{G_i\}_{i=N+1}^{N+M}$.

This problem is difficult because the training data are constructed from past experiments reported in the literature; therefore, they are significantly biased with respect to the uniform distribution over the chemical space

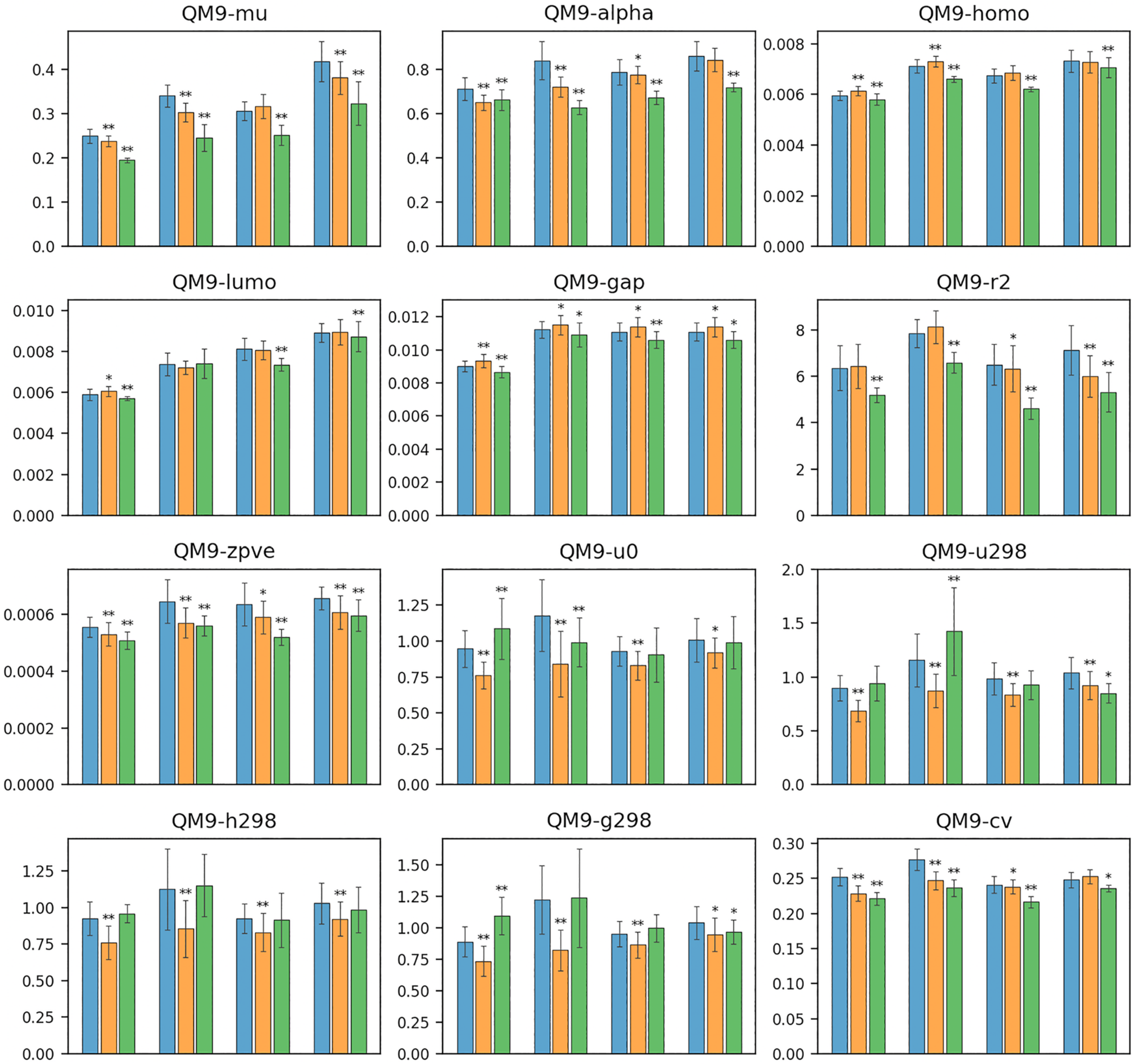

Figure 2. MAE comparison results of QM9. The 12 subplots correspond to 12 property prediction tasks. The x-axises correspond to the four simulated biased sampling scenarios. The blue, orange, and green bars correspond to the Baseline, IPS, and CFR approaches, respectively.

$\mathscr{G}$ because of the decisions implemented by researchers on experimental plans or publication options. Hence, there is no guarantee that the predictor derived from the biased training data has high predictive performance even on the chemical space $\mathscr{G}$.

Note that the $\mathscr{D}^{\text{test}}$ can also be a biased sample; however, without loss of generality, we only assume it is a uniformly random subset of the molecules that we are interested in.

In summary, the inputs and outputs of the problem are as follows:

Input: molecular graph $G_i = (\mathscr{V}_i, \mathscr{E}_i, \sigma_i) \in \mathscr{G}$ in $\mathscr{D}^{\text{train}}$ and $\mathscr{D}^{\text{test}}$ for training and test, respectively.

Output: one target chemical property $\hat{y}_i \in \mathbb{R}$.

The ordinary setting is included in our problem setting when $\mathscr{D}^{\text{train}}$ and $\mathscr{D}^{\text{test}}$ come from the same distribution.

We use Mean Absolute Error (MAE) to evaluate the predictive performance of a model, which is defined as

$$MAE = \frac{1}{M}\sum_{i=1}^{M} |\hat{y}_i - y_i|.$$

Comparison of predictive performance. Figures 2 and Figures 3 show all the MAE comparison results (in the mean and standard deviations of 30 trials) corresponding to the four simulated biased sampling scenarios. The * symbols above the bars denote the *p*-values of the paired *t*-test when comparing Baseline (without

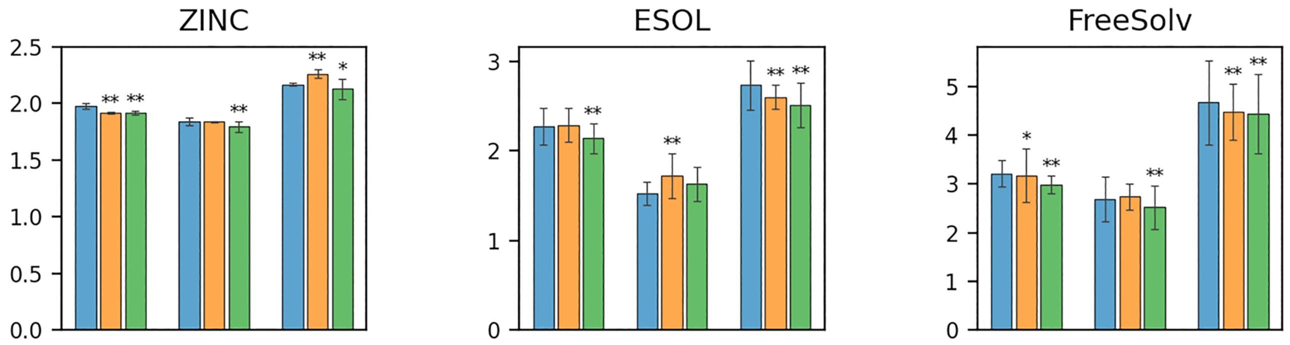

Figure 3. MAE comparison results of ZINC, ESOL, and FreeSolv. The x-axises correspond to the three simulated biased sampling scenarios (Scenario 3 was not performed on them). The blue, orange, and green bars correspond to the Baseline, IPS, and CFR approaches, respectively.

bias mitigation) with IPS and CFR, respectively; ** means the *p*-value was less than 0.01, and * means that it was less than 0.05. Note that, in the case of QM9, the result for the HOMO-LUMO gap (denoted by 'gap') in Scenario 3 was equivalent to that in Scenario 4 because QM9 contains HOMO-LUMO gap information,

The overall results for all the scenarios indicated that the IPS approach improved the performance for many properties and scenarios; in particular the performance was statistically significantly improved for the five properties of QM9 (zvpe, u0, u298, h298, and g298) in all of the four scenarios, and for the three properties of QM9 (mu, alpha, cv) in three out of four scenarios, which indicated that IPS has solid effectiveness and potential in mitigating experimental biases on these tasks. However, we found that there were some statistically insignificant comparison and even significant failure for the four properties of QM9 (homo, lumo, gap, r2) and the properties of ZINC, ESOL, and FreeSolv. These failures indicated that although IPS achieved improvements on most of the tasks, it was not stable. In addition, the improvements by IPS of QM9 were more significant for Scenarios 1 and 2 than Scenarios 3 and 4. The differences might be explained by the accuracy of the propensity score model; the accuracies in the four scenarios were 81.05%, 87.49%, 76.04%, and 79.02%, respectively, which meant that the propensity scores were more accurate in Scenarios 1 and 2.

The CFR approach achieved more remarkable predictive performance than the IPS approach for most of the properties and scenarios. For the four properties of QM9 (homo, lumo, gap, and r2) and the three properties of ZINC, ESOL, and FreeSolv in all of the four scenarios (except for lumo and the property of ESOL in Scenario 2), where IPS failed to improve the predictive performance, CFR achieved statistically significant improvements comparing to the baseline method. Further, for the four properties of QM9 (mu, alpha, zvpe, and cv) in all of the four scenarios (except for alpha in Scenario 1), the improvements achieved by CFR were more remarkable than IPS. However, we found that for the four properties of QM9 (u0, u298, h298, g298), where IPS achieved remarkable improvements, CFR failed to show statistically significant increasing or decreasing of predictive performance comparing to the baseline method, which indicated that CFR was not always effective.

In summary, both of the IPS and CFR approaches made solid improvements in mitigating experimental biases for most of the properties and scenarios, and CFR showed better performance than IPS. However, some failures also indicated that the IPS and CFR approaches were not stable.

Prediction accuracy depending on indicators for biased sampling.

We further investigated why the IPS and CFR approaches successfully corrected the bias by visualizing the prediction accuracy depending on the indicators used in the biased sampling scenarios. Because of space limitations, we only show the predictive performance for the chemical property mu and u0 of QM9 in Figure 4. Similar results can be observed on other tasks. The horizontal axes in the figure correspond to the indicators, namely, (a) the number of atoms, (b) the proportion of double/triple/aromatic bonds, (c) the value of the HOMO-LUMO gap (gap), and (d) the value of the target (i.e., mu for the top and u0 for the bottom), respectively. The top four figures correspond to the average test MAE (that is, the smaller, the more accurate) of predicting mu of QM9 and the bottom four figures correspond to the average test MAE of predicting u0 of QM9. Note that we obtained the average test MAEs in Figure 4 from another 30 trials of biased sampling and model training.

For predicting mu, according to the MAE comparison results, we know that both IPS and CFR achieved significant improvements of predictive performance for most scenarios and CFR outperformed IPS. In Scenario 1, most of the molecules used for training had a number of atoms ranging from 5 to 15. For predicting mu of molecules with 15 or more atoms, primarily in the test dataset, IPS and CFR consistently outperformed the baseline method, and in addition, CFR consistently outperformed IPS. In Scenario 2, most of the molecules used for training had a proportion of double/triple/aromatic bonds higher than 0.4. Again, on those molecules with proportion less than 0.4, IPS and CFR consistently performed better than the baseline method, and CFR almost outperformed IPS. Similarly in Scenarios 3 for predicting mu, we observed the advantage of CFR for the smaller indicator values (less than 0.2) corresponding to the test datasets. However, in Scenario 4, we failed to observe the similar advantages of IPS and CFR for smaller indicator values, which was not consistent with our findings in Figure 2. This failure also indicated that the IPS and CFR approaches lacked stability on some tasks.

For predicting u0, we know that IPS achieved significant improvements of predictive performance for all scenarios while CFR failed. For predicting mu of molecules with 15 or more atoms, IPS consistently outperformed

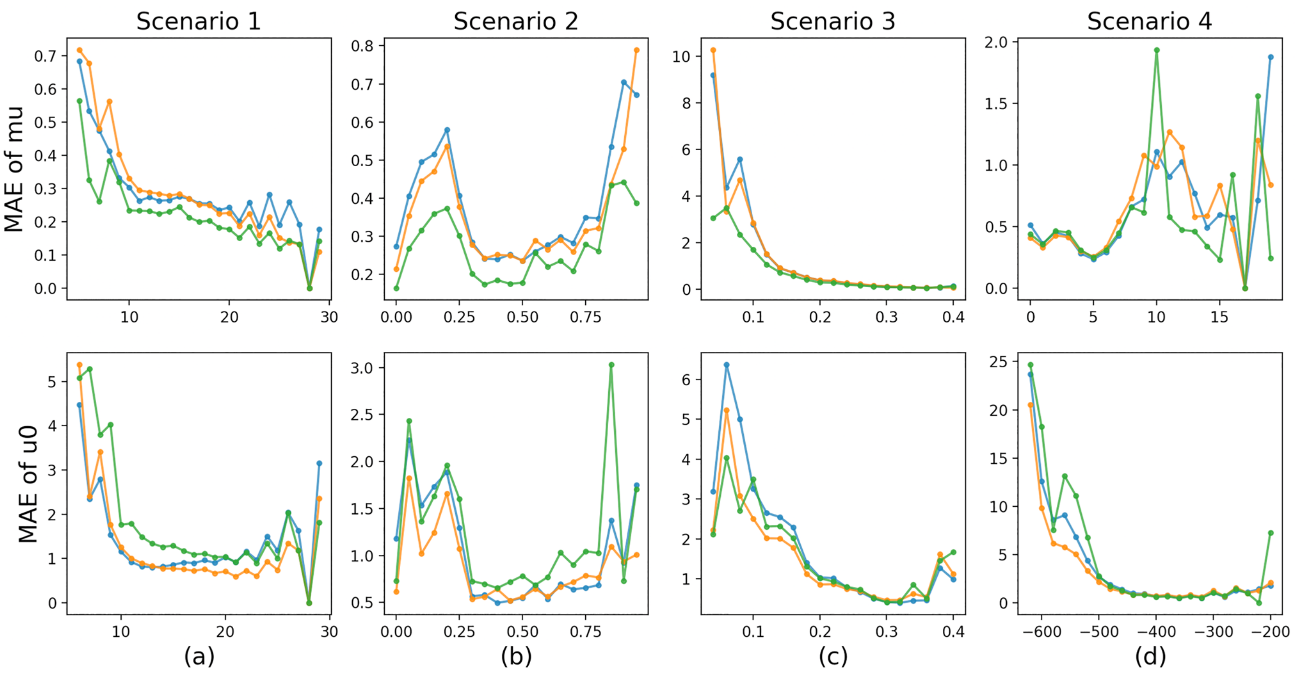

Figure 4. MAE comparison results of mu and u0 depending on indicators of the four biased sampling scenarios. The *x*-axises correspond to (a) number of atoms, (b) proportion of double/triple/aromatic bonds, (c) value of HOMO-LUMO gap (gap), and (d) value of the target (i.e., mu for the top and u0 for the bottom), respectively. The blue, orange and green lines correspond to the Baseline, IPS, and CFR approaches, respectively.

the baseline method while CFR almost failed. In Scenario2, IPS achieved better performance on those molecules with proportion less than 0.4, which was consistent with our findings. However, the performance of CFR was not consistent, which further indicated the low stability of this method on this task. In Scenario 3 and 4 for predicting u0, we observed the advantage of IPS for the smaller indicator values corresponding to the test datasets while we failed to observe the advantage of CFR, which was consistent with our findings.

In summary, by visualizing the prediction accuracy depending on those indicators for biased sampling, we can partially conclude that, on most of the tasks, IPS and CFR improved the predictive performance on molecules with lower chance for observation, which led to the overall improvements.

Discussion

We considered the prediction of chemical properties from datasets that have experimental biases. We introduced two promising bias mitigation techniques by combining the recent developments in causal inference and GNN-based graph learning. We tested four practical biased sampling scenarios on the well-known QM9, ZINC, ESOL, and FreeSolv datasets for experiments. The experimental results confirmed that the two approaches improved the predictive performance in all scenarios on most of the tasks with statistical significance compared with the baseline method, which had no effort for bias mitigation. We also found that the more modern CFR approach outperformed the IPS approach for most of the tasks and scenarios. However, based on our experimental results, the IPS and CFR approaches were not stable enough, which should be further improved. We attribute some of this failure in part to the discussion on transferability and discriminability [52]. When the information that is useful for discriminating the biased and unbiased data is also useful for predicting some chemical properties, it is eliminated by IPS and CFR, resulting in poor discriminability (i.e., prediction power).

Further, as a new study, we would like to describe the future scope in two directions. The fundamental direction of our research is to mitigate experimental biases in scientific research. In this study, we only focused on the population of chemical compounds and the task of property prediction. Possible future topics are extensions to more complex prediction tasks under experimental biases, including chemical-chemical link prediction in the chemical domain, biological network prediction in the biological domain, and discovery of unknown compounds in the material science domain. Another possible direction is applications of more modern and robust methods in covariate shift, causal inference, and domain adaptation. Since the difficulty in our problem setting lies mainly in learning unbiased representation of biased observed instances, it would be promising to adapt general ideas from those related areas and to develop theories and techniques for our specific tasks.

Materials and methods

Datasets. We first used the chemical molecule dataset QM9[48]. QM9 is a publicly available dataset containing 134,000 small stable organic molecules composed of hydrogen (H), carbon (C), oxygen (O), nitrogen (N), and fluorine (F). They are the subset of all the molecules with up to nine heavy atoms (C, O, N, F) out of the 166 billion molecules of the GDB-17 chemical universe[48,51]; therefore, the QM9 dataset can be considered to have a

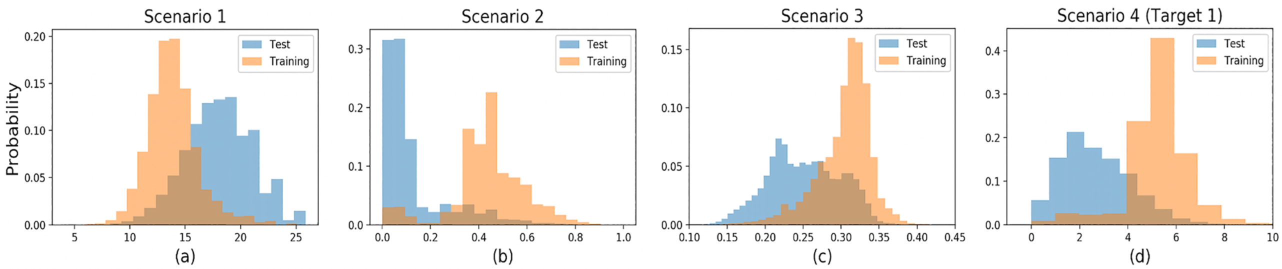

Figure 5. Average densities of the training and test datasets of QM9. The x-axises correspond to (a) number of atoms, (b) proportion of double/triple/aromatic bonds, (c) value of HOMO-LUMO gap (gap), and (d) value of mu, respectively.

close distribution to the natural universe of molecules comprising C, O, N, and F. Each molecule in QM9 has 12 fundamental properties: dipole moment (mu), isotropic polarizability (alpha), HOMO (homo), LUMO (lumo), gap between the homo and lumo (gap), electronic spatial extent (r2), zero point vibrational energy (zpve), internal energy at 0 K (u0), internal energy at 298.15 K (u298), enthalpy at 298.15 K (h298), free energy at 298.15 K (g298), and heat capacity at 298.15 K (cv); they were used as 12 targets for 12 regression tasks.

We further used three chemical molecule datasets, i.e., ZINC [49], ESOL, and FreeSolv [50]. The ZINC dataset contains 250,000 drug-like commercially available molecules graphs with up to 38 heavy atoms. The task of ZINC is to predict a molecular property known as constrained solubility. ESOL is a small dataset consisting of water solubility data for 1,128 molecular graphs. FreeSolv is also a small dataset consisting of 643 molecular graphs with hydration free energy of small molecules in water. Note that molecule structures in these three datasets are far less than exhaustively enumerated. Thus, in contrast with QM9, there is no guarantee that the $\mathscr{D}^{\text{test}}$ of these three datasets can be regarded as an unbiased subset of a chemical space . However, without loss of generality, we only assume the $\mathscr{D}^{\text{test}}$ is unbiased or less biased (compared with the $\mathscr{D}^{\text{train}}$) corresponding to the indicators that we used to induce biases.

Biased sampling scenarios. Because we cannot know why the compounds reported in the literature were selected, we simulated several possible scenarios to sample our training datasets. We considered the following four possible biased sampling scenarios:

- Scenario 1 assumes that molecules with a smaller number of atoms have higher sampling chances, because smaller molecules are considered more common and better explored[22].
- Scenario 2 prefers molecules with smaller proportions of single bonds (i.e., with higher proportions of the other bond types). The spectral manifestations of chemical functional groups significantly depend on bond types within the group[53]. For simplicity, we assumed that less single-bonded molecules would have stronger spectral manifestations.
- Scenario 3 selects molecules with higher values of the gap between the highest energy occupied molecular orbital (HOMO) and lowest energy unoccupied molecular orbital (LUMO), because larger HOMO-LUMO gap values often indicate higher stability and lower chemical reactivity[54].
- Scenario 4 assumes that scientists focus on compounds with high target property values based on their expertise and experience to conduct experiments. Further, compounds with higher targets values have higher sampling chances.

We first sampled a test dataset with a size of 10% of the entire dataset uniformly at random. Then, according to each of the four biased sampling scenarios, we sampled a biased training dataset from the remaining molecules. Each compound had a sampling chance determined by the sigmoid function depending on the corresponding sampling criteria. Take QM9 for example, in Scenario 1, the smallest molecules with three atoms had the largest sampling chances, while the ones with 27 atoms had the smallest chances. The larger the gain of the sigmoid curve becomes, the more the training and test datasets were separated; we tuned the gain to bring the average sampling probability to 10%. We used sampling without replacement; therefore, no graph belonged to the training and test sets simultaneously. We repeated the sampling procedure 30 times to build training and test datasets for statistical testing. Figure 5 shows the average densities of the 30 trials under the biased sampling scenarios of QM9 (for Scenario 4, only the first target among the 12 targets is shown because of the page limitation). The average densities of ZINC, ESOL, and FreeSolv were similar to those in the Scenario 1, 2, and 4 of QM9 in Figure 5. Note that Scenario 3 was not performed on these datasets because HOMO-LUMO gap information is not contained in ZINC, ESOL, and FreeSolv.

Methods.

We begin by reviewing the GNN architecture that is the fundamental building block of our model, and then we describe two bias canceling schemes: IPS and CFR. They are combined to solve the problem of chemical graph property prediction under experimental biases. We summarize the symbols used in this paper in Table 1.

Symbol in PROBLEM SETTING	**Description**
$\mathscr{G}$	A large chemical space
$\mathscr{D}^{\text{train}} = \{(G_i, y_i)\}_{i=1}^{N} \subset \mathscr{G}$	Training dataset of N molecules
$\mathscr{D}^{\text{test}} = \{G_i\}_{i=N+1}^{N+M}$	Test dataset of M molecules
$G_i = (\mathscr{V}_i, \mathscr{E}_i, \sigma_i) \in \mathscr{G}$	Molecular graph
$\mathscr{V}_i$	Set of graph nodes of G_i
$\mathscr{E}_i \subseteq \mathscr{V}_i \times \mathscr{V}_i$	Set of edges of G_i
$\sigma : \mathscr{V}_i \cup \mathscr{E}_i \to \mathscr{L}$	Node and edge label function
$\mathscr{L}$	Set of node and edge labels
$y_i \in \mathbb{R}$	Target chemical property value
Symbol in METHODS	Description
$\mathbf{m}_v^t \in \mathbb{R}^D$	Message of node v in layer t
$\mathbf{h}_v^t \in \mathbb{R}^D$	Feature vector of node v in layer t
$d_i \in \{0, 1\}$	Domain of G_i
$m_t : (\mathbf{h}_v^t, \mathbf{h}_u^t, \sigma(u, v)) \to \mathbb{R}^D$	GNN message function
$a : \mathbb{R}^D \to \mathbb{R}^D$	GNN activation function
$u_t : (\mathbf{h}_v^t, \mathbf{m}_v^t) \to \mathbb{R}^D$	GNN update function
$r : \{\mathbf{h}_v^T\} \to \mathbb{R}^D$	GNN graph-level readout function
$f : \mathscr{G} \to \mathbb{R}$	Property predictor
$f_{\text{F}} : \mathscr{G} \to \mathbb{R}^D$	Feature extractor
$f_{\text{L}} : \mathbb{R}^D \to \mathbb{R}$	Label predictor
$f_{\text{IPM}} : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$	Internal probability metric
$f_{\text{W}} : \mathbb{R}^D \to \mathbb{R}^2$	Weight estimator
$\pi : \mathscr{G} \to [0, 1]$	Propensity score function
$\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^{\geq 0}$	Regression loss function
$c : \{0, 1\} \times [0, 1] \to \mathbb{R}^{\geq 0}$	Classification loss function

Table 1. List of symbols used in this paper.

GNNs to predict chemical properties. Among many successful GNNs, we selected the message-passing GNN architecture proposed by Gilmer et al.[5] owing to its generality, simplicity, and fair performance in the chemical domain.

A GNN uses a graph $G = (\mathcal{V}, \mathcal{E}, \sigma) \in \mathscr{G}$ as its input. In the t-th layer of the GNN, it updates the current set of the node representation vectors $\{\mathbf{h}_v^t\}_{v \in V_i}$ to $\{\mathbf{h}_v^{t+1}\}_{v \in V_i}$. Specifically, the representation vector of node v is updated depending on the current vectors of its neighbor nodes using the following update formula:

$$\mathbf{m}_v^{t+1} = a\left(\sum_{u \in \mathscr{N}(v)} m_t\left(\mathbf{h}_v^t, \mathbf{h}_u^t, \sigma(v, u)\right)\right), \quad \mathbf{h}_v^{t+1} = u_t\left(\mathbf{h}_v^t, \mathbf{m}_v^{t+1}\right),$$

where $\mathscr{N}(v)$ denotes the set of the neighbor nodes of v, and m_t is a so-called message passing function that collects the information (i.e., the representation vectors) of the neighbors, and it is a linear function of $(\mathbf{h}_v^t, \mathbf{h}_u^t)$ depending on the edge label $\sigma(v, u)$. Further, a is an activation function, for which we select the rectified linear unit (ReLU) function, and u_t is the vertex update function, for which we use the gated recurrent unit (GRU). The initial node representations $\{\mathbf{h}_v^0\}_{v \in \mathscr{V}}$ are initialized depending on their atom types. As the message passing operation is repeatedly applied, the node representation vector gradually incorporates information about its surrounding structure.

After being processed through T layers, the final node representations $\{\mathbf{h}_v^T\}_{v \in \mathscr{V}}$ are obtained; they are aggregated to the graph-level representation $\mathbf{h}_G$ using a readout function $\mathbf{h}_G = r\left(\{\mathbf{h}_v^T\}_{v \in \mathscr{V}}\right)$, where we could simply use summation, followed by a linear transformation as the readout function. However, in our implementation, we used a slightly more complex solution: a long short-term memory (LSTM) pooling layer, followed by a linear layer. The graph-level representation $\mathbf{h}_G$ is passed to the final layer to achieve the outputs of the GNN, such as the chemical property, propensity score, and domain weights prediction, as we discuss later.

Bias correction using IPS. If we assume that no biases exist in our training dataset, the distributions of the training and target (test) datasets are identical. This implies that minimizing the empirical mean of the loss function ℓ, that is, $\frac{1}{N}\sum_{i=1}^{N} \ell(y_i, f(G_i))$, directly obtains a good prediction model g that achieves a small expected loss $\mathrm{E}[\ell(y, g(G))]$ for the test data. However, because the training dataset is sampled in a biased manner in our scenario, the minimization of the standard empirical loss results in a biased prediction model.

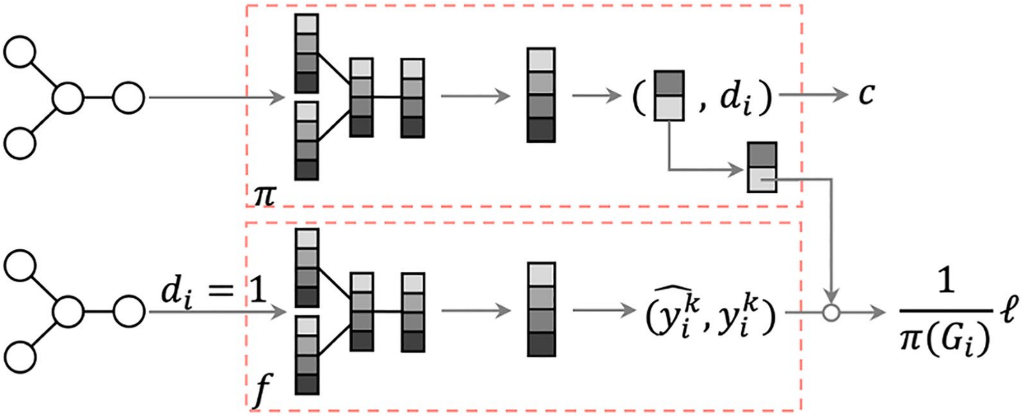

Figure 6. Architecture of the two-step IPS approach.

A possible remedy to this problem is the use of a propensity score[38] to adjust the importance weight of each training instance. The propensity score $\pi(G)$ of molecular graph G is the probability that the molecule is included in the experimental data. The loss for each molecule is inversely weighted with the propensity score, resulting in the modified objective function:

$$o^{\text{IPS}}(f) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\pi(G_i)}\ell(y_i, f(G_i)).$$

With the correct propensity score function π, the weighted loss function is unbiased with respect to the uniform sampling from the chemical space $\mathscr{G}$. As shown in Figure 6, the IPS approach involves two steps: propensity score estimation and chemical property prediction. We first estimated the propensity score function that represents the probability of each molecule being experimented. It is estimated from the biased training dataset and the unbiased test set (uniformly sampled from the chemical space $\mathscr{G}$). This step is frequently performed by solving a two-class probabilistic classification problem to classify the two datasets using the logistic loss (also called the cross-entropy loss). Note that the target property values are not used for propensity modeling. The second step estimates the chemical property prediction model with the loss function weighted using the inverse of the propensity score. We used the squared loss function as ℓ, and the problem is cast as a weighted regression problem. The propensity score function and chemical property prediction model are both implemented as GNNs because they use graph-structured molecules as their inputs.

Bias correction using CFR. The CFR approach is another option to correct sample selection biases. We adopted the method proposed by Hassanpour et al.[47], which is an improvement of the best-known method proposed by Shalit et al.[45]. To apply CFR on our tasks, we introduce $d_i \in \{0, 1\}$ to indicate the treatment (i.e., domain in general) that G_i belongs to (i.e., test or training). The method has four components: a feature extractor, label predictor (replacement of the original treatment outcome predictors), internal probability metric, and weight estimator, denoted by f_{F}, f_{L}, f_{IPM}, and f_{W}, respectively. Different from the original causal effects estimation setting, only the molecular graphs from $\mathscr{D}^{\text{ train}}$ will be passed trough the treatment outcome predictors because molecular graphs from $\mathscr{D}^{\text{ test}}$ have no labels. Thus, there is only one treatment outcome predictor exists in our model for predicting chemical property, which we call label predictor. As shown in Figure 7, in contrast with IPS, which has two separate GNN models, three paths exist in one single model; the first path is for predicting chemical property of $G_i \in \mathscr{D}^{\text{ train}}$, and it is specified by a label predictor f_{L} concatenated after a feature extractor f_{F}. The second path, which takes a batch of molecular graphs $\{G_i\} \subseteq \mathscr{D}^{\text{ train}} \cup \mathscr{D}^{\text{ test}}$ as input, is for measuring distance between distributions, which is specified by internal probability metric f_{IPM} concatenated after the feature extractor f_{F}. The third path is for estimating weight of $G_i \in \mathscr{D}^{\text{ train}} \cup \mathscr{D}^{\text{ test}}$, which is specified by an estimator f_{W} concatenated after the feature extractor f_{F}; Note that f_{F} commonly appears in all of the paths. The final deliverable f that we desire is the composite function of f_{F} and f_{L}; that is, $f(G) = f_{\text{L}}\,(f_{\text{F}}\,(G)$. The weight estimator and the internal probability metric are not our final objective, but they aid in extracting debiased representations of the inputs in the training phase.

In the training phase, the first path (consisting of f_{F} and f_{L}) aims to predict the target chemical property, by minimizing

$$o^{\text{property}}(f_{\text{F}}, f_{\text{L}}) = \frac{1}{N}\sum_{i=1}^{N} w_i \ell(y_i, f_{\text{L}}\,(f_{\text{F}}\,(G_i))),$$

where ℓ is the loss function for chemical properties, namely, the squared loss in our case. Term w_i is the importance sampling weight. According to Hassanpour et al [47], if we denote $\phi_i = f_{\text{W}}\,(f_{\text{F}}\,(G_i)) \in \mathbb{R}^2$ (outputs of two domains) and use softmax function to obtain probabilities, $w_i = \frac{1}{\Pr(d_i|G_i)} = 1 + e^{\phi_i^{(1-d_i)} - \phi_i^{(d_i)}}$ when $N = M$, which

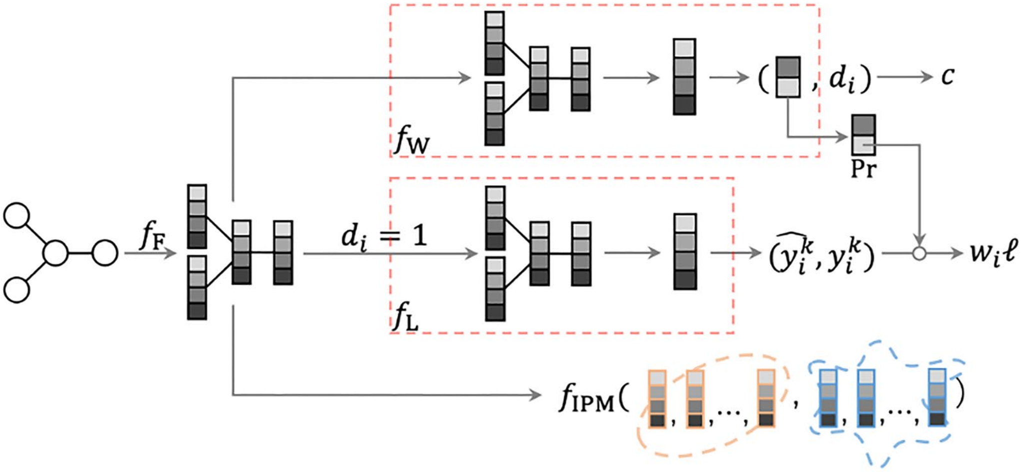

Figure 7. Architecture of the end-to-end CFR approach.

is similar to inverse propensity score in the IPS approach. Note that in our case, d_i is fixed to 1 while calculating w_i because only $G_i \in \mathscr{D}^{\mathrm{train}}$ will be passed through f_L. In the second path (including f_F and f_IPM), the internal probability metric f_IPM aims to measure the distance between distributions of molecular graphs from $\mathscr{D}^{\mathrm{train}}$ and $\mathscr{D}^{\mathrm{test}}$. The objective function for the second path is expressed as

$$o^{\mathrm{IPM}}(f_\mathrm{F}, f_\mathrm{IPM}) = f_\mathrm{IPM}(\{f_\mathrm{F}(G_i)\}_{i:d_i=0}, \{f_\mathrm{F}(G_i)\}_{i:d_i=1}).$$

In the third path (including f_F and f_W), the f_W aims to correctly classify the domain (i.e., test or training) of the input. Denoting the loss function for domain classification (i.e., the cross-entropy loss in our case) by c, the objective function for the third path is expressed as

$$o^{\mathrm{domain}}(f_\mathrm{F}, f_\mathrm{W}) = \frac{1}{N+M}\sum_{i=1}^{N+M} c(d_i, f_\mathrm{W}(f_\mathrm{F}(G_i))).$$

Parameters of f_F, f_L, and f_IPM are updated by minimizing the objective function $o(f_\mathrm{F}, f_\mathrm{L}, f_\mathrm{IPM}) = o^{\mathrm{property}}(f_\mathrm{F}, f_\mathrm{L}) + \alpha \cdot o^{\mathrm{IPM}}(f_\mathrm{F}, f_\mathrm{IPM})$, where α is a hyper-parameter. Note that f_W is set to evaluating mode for calculating $o^{\mathrm{property}}(f_\mathrm{F}, f_\mathrm{L})$ so that there are no gradients accumulated in f_W during this backward process. After the update of f_F, f_L, and f_IPM, f_W is set to training mode, and parameters of f_W are then updated by minimizing $o^{\mathrm{domain}}(f_\mathrm{F}, f_\mathrm{W})$.

Implementation details. We used the PyTorch Geometric library[55] to implement the GNN models. In QM9 and ZINC, each atom is encoded as a 13 and 28-dimensional vector (one hot), respectively, depending on the atom type. In ESOL and FreeSolv, we followed the settings in MoleculeNet [50], where each atom is encode as a 9-dimensional vector. The message passing function m_t depends on edge types and is 32-dimensional. The update function u_t is a gated recurrent unit with 32 internal dimensions. The readout function r is a sequence-to-sequence layer followed by two linear layers with 32 internal dimensions.

The IPS approach required two GNN models, one for the propensity score function and the other for chemical property prediction. In the former, we used $T = 3$ GNN layers and a logistic function as the final layer because the propensity score indicated the probability that a chemical compound is observed. We used the ADAM[56] optimizer with no weight decay and a batch size of 64 for each iteration. We used a learning rate updating scheduler with an initial learning rate of $1e-5$; this reduced the learning rate by a factor of 0.7 until the validation error stopped reducing for five training epochs. The validation datasets were randomly selected to include 20% of the training and test sets. The optimized model that achieved the lowest validation error on the validation dataset was applied to infer the importance. The chemical property prediction models also had $T = 3$ GNN layers, and the other training settings were almost the same as those for the propensity score model. The validation set was 20% of the training set. Because we had 15 target chemical properties, we trained 15 different GNN predictors.

The network structure for the CFR approach had three output paths: the label predictor, internal probability metric, and weight estimator. The label predictor and weight estimator shared the common feature extraction layers on the input side. We set the number of the GNN layers corresponding to the feature extractor and those for the weight estimator and label predictor to 3. Similar to the IPS approach, the weight estimator had a readout function for classification and the label predictor had one for regression. Note that the feature extractor had no readout function. We used Wasserstein distances[57] for the internal probability metric. In addition, before the internal probability metric, there was readout function to aggregate features extracted by the feature extractor to a batch of graph-level features. We set the α for QM9, ZINC, ESOL, and FreeSolv to 10, 10, 100, and 10, respectively. As with the IPS approach, we used ADAM with no weight decay as the optimizer. We also used a learning rate updating scheduler, with an initial learning rate of $1e-5$ and reduced the learning rate by a factor of 0.7 until the validation error stopped reducing for five training epochs. The batch size was set to 64. In contrast with IPS, the entire network was trained in an end-to-end manner. We trained 15 GNNs for each of the target properties in each trial. 20% of the training and test sets were used to validate the domain classifier, and 20% of the training set was used for the label predictor.

As the baseline method, we used the same GNN structure as the one for IPS, but without bias mitigation. In contrast to the IPS approach, we used the standard unweighted average loss for the training dataset. The same settings as for IPS were used except for those specific to IPS, such as the number of GNN layers, the selection of the hyperparameters, and the training and validation sets.

Data availability

The datasets used and/or analysed during the current study are all available at https://pytorch-geometric.readthedocs.io/en/latest/modules/datasets.html. We further processed these datasets with biased sampling scenarios, which were introduced in Material and Methods section. The processed datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Received: 9 December 2021; Accepted: 26 April 2022
Published online: 17 May 2022

References

1. Rupp, M., Tkatchenko, A., Müller, K.-R. & Von Lilienfeld, O. A. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.* **108**, 058301 (2012).
2. Hamilton, W.L., Ying, R. & Leskovec, J. Representation learning on graphs: Methods and applications. IEEE Data Eng. Bull. (2017).
3. Wu, Z. *et al.* A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems* (2020).
4. Duvenaud, D. K. *et al.* Convolutional networks on graphs for learning molecular fingerprints. *In Adv. Neural Inf. Process. Syst.* **28**, 2224–2232 (2015).
5. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O. & Dahl, G.E. Neural message passing for quantum chemistry. In *ICML*, 1263–1272 (2017).
6. Kearnes, S., McCloskey, K., Berndl, M., Pande, V. & Riley, P. Molecular graph convolutions: Moving beyond fingerprints. *J. Comput. Aided Mol. Design* **30**, 595–608 (2016).
7. Veličković, P. *et al.* Graph attention networks. In *ICLR* (2018).
8. Hamilton, W., Ying, Z. & Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.*, 1024–1034 (2017).
9. Li, R., Wang, S., Zhu, F. & Huang, J. Adaptive graph convolutional neural networks. arXiv:1801.03226 (2018).
10. Xu, K., Hu, W., Leskovec, J. & Jegelka, S. How powerful are graph neural networks? In *ICLR* (2018).
11. Liu, Q., Allamanis, M., Brockschmidt, M. & Gaunt, A. Constrained graph variational autoencoders for molecule design. *In Adv. Neural Inf. Process. Syst.* **31**, 7795–7804 (2018).
12. You, J., Liu, B., Ying, Z., Pande, V. & Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. *In Adv. Neural Inf. Process. Syst.* **31**, 6410–6421 (2018).
13. De Cao, N., & Kipf, T. An implicit generative model for small molecular graphs. In *DGMs*, Molgan, (2018).
14. Ying, Z., Bourgeois, D., You, J., Zitnik, M. & Leskovec, J. GNNExplainer: Generating explanations for graph neural networks. *Adv. Neural Inf. Process. Syst.* **32**, 9244–9255 (2019).
15. Akita, H. *et al.* Bayesgrad: Explaining predictions of graph convolutional networks. In *ICONIP*, 81–92 (2018).
16. Harada, S. *et al.* Dual graph convolutional neural network for predicting chemical networks. *BMC Bioinform.* **21**, 1–13 (2020).
17. Wang, H., Lian, D., Zhang, Y., Qin, L. & Lin, X. Gognn: Graph of graphs neural network for predicting structured entity interactions. In *IJCAI* (2020).
18. Llinas, A., Burley, J. C., Box, K. J., Glen, R. C. & Goodman, J. M. Diclofenac solubility: Independent determination of the intrinsic solubility of three crystal forms. *J. Med. Chem.* **50**, 979–983 (2007).
19. Raymer, B. & Bhattacharya, S. K. Lead-like drugs: A perspective: Miniperspective. *J. Med. Chem.* **61**, 10375–10384 (2018).
20. Hann, M. M. Molecular obesity, potency and other addictions in drug discovery. *Med. Chem. Commun.* **2**, 349–355 (2011).
21. Jia, X. *et al.* Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis. *Nature* **573**, 251–255 (2019).
22. Lipinski, C. A. Lead-and drug-like compounds: The rule-of-five revolution. *Drug Discov. Today Technol.* **1**, 337–341 (2004).
23. Hattori, K., Wakabayashi, H. & Tamaki, K. Predicting key example compounds in competitors' patent applications using structural information alone. *J. Chem. Inf. Model.* **48**, 135–142 (2008).
24. Walker, R. *et al.* *Applications of Reference Materials in Analytical Chemistry* (2001).
25. Kearnes, S., Goldman, B. & Pande, V. Modeling industrial ADMET data with multitask networks. arXiv:1606.08793 (2016).
26. Wallach, I. & Heifets, A. Most ligand-based classification benchmarks reward memorization rather than generalization. *J. Chem. Inf. Model.* **58**, 916–932 (2018).
27. Chen, G. *et al.* Alchemy: A quantum chemistry dataset for benchmarking ai models. arXiv:1906.09427 (2019).
28. Kovács, D. P., McCorkindale, W. & Lee, A. A. Quantitative interpretation explains machine learning models for chemical reaction prediction and uncovers bias. *Nat. Commun.* **12**, 1–9 (2021).
29. Quionero-Candela, J., Sugiyama, M., Schwaighofer, A. & Lawrence, N.D. *Dataset Shift in Machine Learning* (The MIT Press, 2009).
30. Pan, S. J. & Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**, 1345–1359 (2009).
31. Ganin, Y. *et al.* Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**, 2096–2030 (2016).
32. Tzeng, E., Hoffman, J., Saenko, K. & Darrell, T. Adversarial discriminative domain adaptation. In *CVPR*, 7167–7176 (2017).
33. Tang, H. & Jia, K. Discriminative adversarial domain adaptation. In *AAAI*, 5940–5947 (2020).
34. Tanwani, A.K. Domain-invariant representation learning for sim-to-real transfer. *arXiv preprint*arXiv:2011.07589 *(2020).*
35. Long, M., Cao, Z., Wang, J. & Jordan, M.I. Conditional adversarial domain adaptation. *arXiv preprint*arXiv:1705.10667 *(2017).*
36. Lee, S., Kim, D., Kim, N. & Jeong, S.-G. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *ICCV*, 91–100 (2019).
37. Ma, X., Zhang, T. & Xu, C. Gcan: Graph convolutional adversarial network for unsupervised domain adaptation. In *CVPR*, 8266–8276 (2019).
38. Imbens, G.W. & Rubin, D.B. *Causal inference in statistics, social, and biomedical sciences* (2015).
39. Schnabel, T., Swaminathan, A., Singh, A. & Chandak, N., & Joachims, T. Recommendations as treatments: Debiasing learning and evaluation. In *ICML* (2016).
40. Ma, W. & Chen, G.H. Missing not at random in matrix completion: The effectiveness of estimating missingness probabilities under a low nuclear norm assumption. *arXiv preprint*arXiv:1910.12774 *(2019).*

41. Zhang, G. *et al.* Selection bias explorations and debias methods for natural language sentence matching datasets. *arXiv preprint*arXiv:1905.06221 *(2019).*
42. Eichler, H.-G. *et al.* Threshold-crossing: A useful way to establish the counterfactual in clinical trials?. *Clin. Pharmacol. Therapeutics* **100**, 699–712 (2016).
43. LaLonde, R.J. Evaluating the econometric evaluations of training programs with experimental data. *The American economic review* 604–620 (1986).
44. Zhao, S. & Heffernan, N. Estimating individual treatment effect from educational studies with residual counterfactual networks. *International Educational Data Mining Society* (2017).
45. Shalit, U., Johansson, F.D. & Sontag, D. Estimating individual treatment effect: generalization bounds and algorithms. In *ICML*, 3076–3085 (2017).
46. Yao, L. *et al.* Representation learning for treatment effect estimation from observational data. *In Adv. Neural Inf. Process. Syst.* **31**, 2633–2643 (2018).
47. Hassanpour, N. & Greiner, R. Counterfactual regression with importance sampling weights. In *IJCAI*, 5880–5887 (2019).
48. Ramakrishnan, R., Dral, P. O., Rupp, M. & Von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data* **1**, 140022 (2014).
49. Gómez-Bombarelli, R. *et al.* Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Sci.* **4**, 268–276 (2018).
50. Wu, Z. *et al.* Moleculenet: A benchmark for molecular machine learning. *Chem. Sci.* **9**, 513–530 (2018).
51. Ruddigkeit, L., Van Deursen, R., Blum, L. C. & Reymond, J.-L. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *J. Chem. Inf. Model.* **52**, 2864–2875 (2012).
52. Chen, X., Wang, S., Long, M. & Wang, J. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *ICML*, 1081–1090 (2019).
53. Sousa-Silva, C., Petkowski, J. J. & Seager, S. Physical chemistry chemical physics. *Phys. Chem. Chem. Phys.* **21**, 18970–18987 (2019).
54. Aihara, J. Reduced HOMO-LUMO gap as an index of kinetic stability for polycyclic aromatic hydrocarbons. *J. Phys. Chem. A* **103**, 7487–7495 (1999).
55. Fey, M. & Lenssen, J.E. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds* (2019).
56. Kingma, D.P. & Ba, J. Adam: a method for stochastic optimization. arXiv:1412.6980 (2014).
57. Cuturi, M. & Doucet, A. Fast computation of wasserstein barycenters. In *ICML*, 685–693 (2014).

Acknowledgements

We are grateful to Yoshihiro Yamanishi, Ichhigaku Takigawa, and Shonosuke Harada for valuable information and discussions. This work is partially supported by JSPS KAKENHI GrantNumber 20H00609.

Author contributions

Y.L. and H.K. contributed to the study design. Y.L. conducted the experiments and analyzed the results. H.K. gave technical support and conceptual advice for the methodology. The first draft of the manuscript was written by Y.L., and the revised manuscript was written by H.K. All authors reviewed and approved the final manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.