EXPANDING GENOMIC DISCOVERY: CAUSALLY-INSPIRED NEURAL NETWORKS FOR PREDICTING THERAPEUTIC TARGETS

Guadalupe Gonzalez
Imperial College London, London, UK
Prescient Design, Genentech, SSF, CA, USA
gonzalez.guadalupe@gene.com

Isuru Herath
Cornell University, Ithaca, NY, USA
Merck & Co., SSF, CA, USA
ish9@cornell.edu

Kirill Veselkov
Imperial College London, London, UK
kirill.veselkov04@imperial.ac.uk

Michael Bronstein
University of Oxford, Oxford, UK
michael.bronstein@gmail.com

Marinka Zitnik
Harvard Medical School, Cambridge, MA, USA
Kempner Institute for the Study of Natural and Artificial Intelligence, Harvard University, Cambridge, MA, USA
Broad Institute of MIT and Harvard, Cambridge, MA, USA
Harvard Data Science Initiative, Cambridge, MA, USA
marinka@hms.harvard.edu

ABSTRACT

As an alternative to target-driven drug discovery, phenotype-driven approaches identify compounds that counteract the overall disease effects by analyzing phenotypic signatures. Our study introduces a novel approach to this field, aiming to expand the search space for new therapeutic agents. We introduce PDGRAPHER, a causally-inspired graph neural network model designed to predict arbitrary perturbagens – sets of therapeutic targets – capable of reversing disease effects. Unlike existing methods that learn responses to perturbations, PDGRAPHER solves the inverse problem, which is to infer the perturbagens necessary to achieve a specific response – i.e., directly predicting perturbagens by learning which perturbations elicit a desired response. Experiments across eight datasets of genetic and chemical perturbations show that PDGRAPHER successfully predicted effective perturbagens in up to 9% additional test samples and ranked therapeutic targets up to 35% higher than competing methods. A key innovation of PDGRAPHER is its direct prediction capability, which contrasts with the indirect, computationally intensive models traditionally used in phenotype-driven drug discovery that only predict changes in phenotypes due to perturbations. The direct approach enables PDGRAPHER to train up to 30 times faster, representing a significant leap in efficiency. Our results suggest that PDGRAPHER can advance phenotype-driven drug discovery, offering a fast and comprehensive approach to identifying therapeutically useful perturbations.

1 INTRODUCTION

Target-driven drug discovery, predominant since the 1990s, focuses on the design of highly specific compounds against disease-associated targets such as proteins or enzymes (Vincent et al., 2022; Moffat et al., 2017). Despite numerous successful examples (Druker et al., 1996; Bange et al., 2001), the past decade has seen a revival of phenotype-driven approaches in an attempt to go beyond the "one drug, one gene, one disease" model of target-driven approaches. Phenotype-driven drug discovery aims to identify compounds or, more broadly, perturbagens – combinations of therapeutic targets – that reverse phenotypic disease effects as measured by high-throughput phenotypic assays

such as cellular responses, as represented by gene expression profiles (Musa et al., 2018; Vincent et al., 2022). Recent advances in deep learning produced methods that predict gene expression responses to perturbagens or combinations thereof (Zhu et al., 2021; Pham et al., 2021; Lotfollahi et al., 2019; Hetzel et al., 2022). Such methods have advanced lead discovery by enabling predictions of responses to perturbagens that were not yet experimentally tested. 1) However, current approaches rely on predefined perturbagen libraries, meaning that they can select perturbagens only from predefined libraries instead of flexibly identifying perturbagens as combinations of therapeutic targets. 2) Existing approaches are predominantly perturbation response methods that predict changes in phenotypes upon perturbations. Thus, they only indirectly identify perturbagens by exhaustively predicting responses to all perturbations in the library and then searching for perturbagens with the desired response. 3) Unlike existing methods that learn responses to perturbations, phenotype-driven drug discovery needs to solve the inverse problem, which is to infer perturbagens necessary to achieve a specific response – i.e., directly predicting perturbagens by learning which perturbations elicit a desired response. In causal discovery, the problem of identifying which elements of a

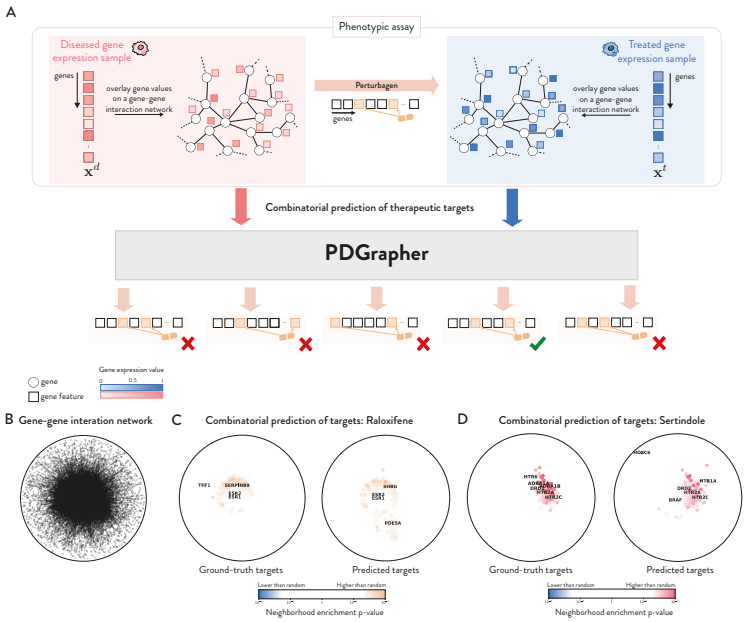

Figure 1: **Overview of PDGRAPHER**. **(A)** Given a paired diseased and treated gene expression samples, and a proxy causal graph, PDGRAPHER predicts a candidate set of therapeutic targets to shift cell gene expression from diseased to treated state. **(B-D)** Depicted is the PPI we use throughout our work using SAFE (Baryshnikova, 2016) (B). Spatial enrichment of ground-truth and predicted gene targets for Raloxifene (C) and Sertindole (D) in Chemical-PPI-Lung as computed by SAFE show high overlap. More details can be found in Appendix A.1.

system should be perturbed to achieve a desired state is referred to as *optimal intervention design* (Hauser & Bühlmann, 2014; Ghassami et al., 2018; Agrawal et al., 2020). Leveraging insights from causal discovery and geometric deep learning, here we introduce PDGRAPHER, a novel approach for combinatorial prediction of therapeutic targets that can shift gene expression from an initial, diseased state to a desired treated state. PDGRAPHER is formulated using a causal model, where genes represent the nodes in a causal graph, and structural causal equations define their causal relationships. Given a genetic or chemical intervention dataset, PDGRAPHER pinpoints a set of genes that a perturbagen should target to facilitate the transition of node states from diseased to treated (Figure 1A). PDGRAPHER utilizes protein-protein interaction networks (PPI) and gene regulatory networks (GRN) as approximations of the causal graph, operating under the assumption of no unobserved confounders. PDGRAPHER tackles the optimal intervention design objective using representation learning, utilizing a graph neural network (GNN) to represent the structural equations. We evaluate PDGRAPHER across eight datasets, comprising genetic and chemical interventions across two cancer types and proxy causal graphs, and consider diverse evaluation setups, including settings

where held out folds contain novel samples and challenging settings where held out folds contain novel samples from a cancer type that PDGRAPHER has never encountered before (Figure 4BC). Our experiments show PDGRAPHER's superior performance in the combinatorial prediction of therapeutic targets compared to mechanistic and response prediction baselines. Additionally, we show PDGRAPHER's predictions follow network proximity principles that govern gene similarities, and that PDGRAPHER can illuminate mode of action of existing chemical perturbagens. When trained, PDGRAPHER predicts perturbagens (as a set of candidate target genes) to shift cells from diseased to treated. An example of PDGRAPHER's predictions is depicted in Figure 1B-D where we observe consistent patterns between ground truth and predicted gene targets and their spatial enrichment distributions. Our work integrates deep learning with causal inference to advance phenotype-driven drug discovery. More details on related work can be found in Appendix A.2.

2 METHODOLOGY

Problem formulation - combinatorial prediction of therapeutic targets. Intuitively, given a diseased cell line sample, we would like to predict the set of therapeutic genes that need to be targeted to reverse the effects of disease, that is, the genes that need to be perturbed to shift the cell gene expression state as close as possible to the healthy state. Next, we formalize our problem formulation. Let $\mathtt{M} =< \mathbf{E}, \mathbf{V}, \mathcal{F}, P(\mathbf{E}) >$ be a structural causal model (SCM) associated with causal graph G, where $\mathbf{E}$ is a set of exogenous variables affecting the system, $\mathbf{V}$ are the system variables, $\mathcal{F}$ are structural equations encoding causal relations between variables and $P(\mathbf{E})$ is a probability distribution over exogenous variables. Let $\mathcal{T} = \{\mathtt{T}_1, ..., \mathtt{T}_m\}$ be a dataset of paired healthy and diseased samples, where each element is a 3-tuple $\mathtt{T} =< \mathbf{v}^h, \mathbf{U}, \mathbf{v}^d >$ with $\mathbf{v}^h \in [0,1]^N$ being gene expression values of healthy cell line (variable states before perturbation), $\mathbf{V_U}$ being the disease-causing perturbed variable (gene) set in $\mathbf{V}$, and $\mathbf{v}^d \in [0,1]^N$ being gene expression values of diseased cell line (variable states after perturbation). Our goal is to find, for each sample $\mathtt{T} =< \mathbf{v}^h, \mathbf{U}, \mathbf{v}^d >$, the variable set $\mathbf{U'}$ with the highest likelihood of shifting variable states from diseased $\mathbf{v}^d$ to healthy $\mathbf{v}^h$ state. To increase generality, we refer to the desired variable states as *treated* ($\mathbf{v}^t$). Our goal can then be expressed as

$$argmax_{\mathbf{U'}} P^{G^{\mathbf{U}}}(\mathbf{V} = \mathbf{v}^t \mid do(\mathbf{U'})), \tag{1}$$

where $P^{G^{\mathbf{U}}}$ represents the probability over graph G mutilated upon perturbations in variables in $\mathbf{U}$. Under the assumption of no unobserved confounders, the above interventional probability can be expressed as a conditional probability on the mutilated graph $G^{\mathbf{U'}}$:

$$argmax_{\mathbf{U'}} P^{G^{\mathbf{U'}}}(\mathbf{V} = \mathbf{v}^t \mid \mathbf{U'}), \tag{2}$$

which under the causal Markov condition is:

$$argmax_{\mathbf{U'}} \prod_i P(\mathrm{v}V_i = \mathrm{v}v_i^t \mid \mathbf{Pa}_{\mathrm{v}V_i}), \tag{3}$$

where $\mathbf{Pa}_{\mathrm{v}V_i}$ represents parents of variable $\mathrm{v}V_i$ according to graph $G^{\mathbf{U'}}$ (that is, the mutilated graph upon intervening on variables in $\mathbf{U'}$). Here, state of a variable $\mathrm{v}V_j \in \mathbf{Pa}_{\mathrm{v}V_i}$ will be equal to an arbitrary value $\mathrm{v}v_j'$ if $\mathrm{v}V_j \in \mathbf{U'}$. Therefore, intervening on the variable set $\mathbf{U'}$ modifies the graph used to obtain conditional probabilities and determines the state of variables in $\mathbf{U'}$.

Problem formulation - representation-learning-based combinatorial prediction of therapeutic targets. In the previous section, we drew on the SCM framework to introduce a general formulation for the task of combinatorial prediction of therapeutic targets. Instead of approaching the problem from a purely causal inference perspective, we draw upon representation learning to approximate the queries of interest to address the limiting real-world setting of a noisy and incomplete causal graph. Formulating our problem using the SCM framework allows for explicit modeling of interventions and formulation of interventional queries. Inspired by this principled problem formulation, we next introduce the problem formulation using a representation learning paradigm.

We let $G = (\mathcal{V}, \mathcal{E})$ denote a graph with $|\mathcal{V}| = n$ nodes and $|\mathcal{E}|$ edges, which contains partial information on causal relationships between nodes in $\mathcal{V}$ and some noisy relationships. We refer to this graph as *proxy causal graph*. Let $\mathcal{T} = \{\mathtt{T}_1, ..., \mathtt{T}_{\mathrm{v}M}\}$ be a dataset with an individual sample being a 3-tuple $\mathtt{T} =< \mathbf{x}^h, \mathcal{U}, \mathbf{x}^d >$ with $\mathbf{x}^h \in [0,1]^n$ being the node states (attributes) of healthy

cell sample (before perturbation), $\mathcal{U}$ being the set of disease-causing perturbed nodes in $\mathcal{V}$, and $\mathbf{x}^d \in [0,1]^n$ being the node states (attributes) of diseased cell sample (after perturbation). We denote by $G^{\mathcal{U}} = (\mathcal{V}, \mathcal{E}^{\mathcal{U}})$ the graph resulting from the mutilation of edges in G as a result of perturbing nodes in $\mathcal{U}$ (one graph per perturbagen; we avoid using superindices for simplicity). Here again, we refer to the desired variable states as *treated* ($\mathbf{x}^t$). Our goal is then to learn a function:

$$f : G^{\mathcal{U}'},\ \mathbf{x}^d, \mathbf{x}^t \rightarrow argmax_{\mathcal{U}'} P^{G^{\mathcal{U}'}}(\mathbf{x} = \mathbf{x}^t | \mathbf{x}^d, \mathcal{U}') \tag{4}$$

That, given the graph $G^{\mathcal{U}'}$, the diseased $\mathbf{x}^d$ and treated $\mathbf{x}^t$ node states, predicts the combinatorial set of nodes $\mathcal{U}'$ that if perturbed have the highest chance of shifting the node states to the treated state $\mathbf{x}^t$. We note here that $P^{G^{\mathcal{U}'}}$ represents probabilities over graph $G^{\mathcal{U}}$ mutilated upon perturbations in nodes in $\mathcal{U}'$. Under Causal Markov Condition, we can factorize $P^{G^{\mathcal{U}'}}$ over graph $G^{\mathcal{U}'}$:

$$f : G^{\mathcal{U}'},\ \mathbf{x}^d, \mathbf{x}^t \rightarrow argmax_{\mathcal{U}'} \prod_i P(\mathbf{x}_i = \mathbf{x}_i^t | \mathbf{x}_{\mathcal{PA}_i}) \tag{5}$$

that is, the probability of each node depending only on its parents $\mathcal{PA}_i$ in graph $G^{\mathcal{U}'}$.

We assume (i) real-valued node states, (ii) G is fixed and given, and (iii) atomic and non-atomic perturbagens (intervening on individual nodes or sets of nodes). Given that the value of each node should depend only on its parents on the graph $G^{\mathcal{U}'}$, a message-passing framework appears especially suited to compute the factorized probabilities P. We comment on the problem formulation and its similarity to canonical graph prediction tasks in Appendix A.4.

Intuitively, an exhaustive approach to solving Equation 5 would be to search the space of all potential sets of therapeutic targets $\mathcal{U}'$ and score how effective each is in achieving the desired treated state. This is, indeed, how many cell response prediction approaches can be used for perturbagen discovery (Hetzel et al., 2022; Lotfollahi et al., 2019; 2023). However, with moderately sized graphs, this is highly computationally expensive, if not intractable. Instead, we propose to search for potential perturbagens efficiently with a 2-module approach. First, a perturbagen discovery module f_p searches the space of potential gene sets to predict a suitable candidate $\mathcal{U}'$. Next, a response prediction module f_r checks the goodness of the predicted set $\mathcal{U}'$, that is, how effective intervening on variables in $\mathcal{U}'$ is to shift node states to the desired treated state $\mathbf{x}^t$.

$$(1)\ \ \mathbf{x}^d, \mathbf{x}^t \xrightarrow{f_p} \hat{\mathcal{U}}'$$

$$(2)\ \ \mathbf{x}^d, \hat{\mathcal{U}}' \xrightarrow{f_r} \hat{\mathbf{x}}^t$$

Model optimization. We optimize our response prediction module f_r using cross-entropy loss on known triplets $< \mathbf{x}^h, \mathcal{U}, \mathbf{x}^d >$ and $< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$:

$$\mathcal{L}_{f_r} = CE\left(\mathbf{x}^d,\ f_r(\mathbf{x}^h,\ \mathcal{U})\right) + CE\left(\mathbf{x}^t,\ f_r(\mathbf{x}^d,\ \mathcal{U}')\right)$$

We optimize our intervention discovery module f_p using a cycle loss such that the response upon a predicted $\mathcal{U}'$ is as close to the desired treated state as possible. In addition, we provide a supervisory signal for predicting $\mathcal{U}'$ in the form of cross-entropy loss.

$$\mathcal{L}_{f_p} = CE\left(\mathbf{x}^t,\ f_r(\mathbf{x}^d,\ f_p(\mathbf{x}^d,\ \mathbf{x}^t))\right) + CE\left(\mathcal{U}', f_p(\mathbf{x}^d,\ \mathbf{x}^t)\right)\ \ (with\ f_r\ frozen)$$

We train f_p and f_r in parallel and implement early stopping separately (see Appendix A.9 for more details). Trained modules f_p and f_r are then used to predict, for each diseased cell sample, which nodes should be perturbed ($\mathcal{U}'$) to achieve a desired treated state (Figure 1A). Both f_p and f_r and GNN-based models that model interventions as mutilations in the proxy causal graphs. More details can be found in Appendix A.5.

3 DATASETS, AND EVALUATION

A description of the data sources and preprocessing steps can be found in Appendix A.6. We built datasets comprising gene expression measurements from healthy, diseased, and treated cell lines

to study disease and treatment interventions. We have a total of eight datasets across two treatment types (genetic and chemical interventions), two cancer types (lung cancer cell line A549 and breast cancer cell line MCF7), and two proxy causal graphs (PPI, and GRN), which we denote as follows: Genetic-PPI-Lung, Genetic-PPI-Breast, Chemical-PPI-Lung, Chemical-PPI-Breast, Genetic-GRN-Lung, Genetic-GRN-Breast, Chemical-GRN-Lung, and Chemical-GRN-Breast. Genetic interventions are single-gene knockout experiments by CRISPR/Cas9-mediated gene knockouts, while chemical interventions are multiple-gene treatments induced using chemical compounds. Each dataset is made up of disease and treatment intervention data. Disease intervention data contains paired healthy and diseased gene expression samples and disease-associated genes. Treatment intervention data contains paired diseased and treated gene expression samples and genetic or chemical perturbagens. More details on intervention data can be found in Appendix A.7. Figure 3 summarizes the number of samples for each cell line and intervention dataset type. We benchmark the performance of PDGRAPHER against a set of baselines: Random baseline, Cancer genes, Cancer drug targets, and scGen (Lotfollahi et al., 2019). More details can be found in Appendix A.8. We evaluate PDGRAPHER and baseline methods on a random and a leave-cell-line-out split, using 5-fold cross-validation (Figure 4BC). More details on splits, evaluation setup and metrics can be found in Appendix A.9.

4 RESULTS

PDGRAPHER efficiently predicts genetic and chemical perturbagens to shift cells from diseased to treated states. Given pairs of diseased and treated samples, PDGRAPHER is trained to output a ranking of genes, with the top-predicted genes identified as candidate combinatorial therapeutic targets to shift gene expression phenotype from a diseased to a treated state in each sample. In held out folds that contain novel samples, PDGRAPHER ranks ground-truth therapeutic targets up to 34% higher in chemical intervention datasets and 16% higher in genetic intervention datasets than existing methods (Table 1). Even in held-out folds containing novel samples from a previously unseen disease, PDGRAPHER maintains robust performance (Table 4). Because perturbagens target multiple genes, we measure the fraction of samples in the test set for which we obtain a partially accurate prediction, where at least one of the predicted gene targets corresponds to an actual gene target. PDGRAPHER consistently provides accurate predictions for more samples in the test set than baselines (Tables 1, 4). We observe consistently strong performance of PDGRAPHER across chemical and genetic intervention datasets in the random and leave-cell-out setting using GRNs as the proxy causal graph (Figure 7).

A key innovative feature of PDGRAPHER is its direct prediction of perturbagens that can shift gene expression from diseased to treated states in contrast with existing methods that indirectly predict perturbagens through extensive computational modeling of cell responses (Figure 4A). This feature of PDGRAPHER enables model training up to 30 times faster than indirect prediction methods like scGen (Lotfollahi et al., 2019) (Table 1). We also find that in chemical intervention datasets, candidate therapeutic targets predicted by PDGRAPHER are closer to ground-truth therapeutic targets in the gene-gene interaction network than what would be expected by chance (Figure 5). This result implies that PDGRAPHER not only identifies relevant gene targets but does so in a manner that reflects the underlying biological and network-based relationships (Kamimoto et al., 2023), suggesting that its predictions are rooted in the inherent structure of the gene interaction network which governs gene similarity (Barabási et al., 2011; Ruiz et al., 2021; Eyuboglu et al., 2023).

PDGRAPHER illuminates mode of action of chemical perturbagens. We demonstrate PDGRAPHER's capability to elucidate therapeutic perturbagens' mechanisms of action through the analysis of Raloxifene and Sertindole effects within the Chemical-PPI-Lung dataset (Figure 2). Utilizing network visualization tools, we visually represent the predicted therapeutic targets and their interaction communities, revealing PDGRAPHER's accuracy in predicting known and potentially novel targets for both drugs. Raloxifene's analysis highlights PDGRAPHER's ability to predict its established targets (ESR1, ESR2) and suggests novel targets (SHBG, PDE5A) that align with known physiological effects, offering insights into Raloxifene's broader impact on estrogen-related pathways. Similarly, for Sertindole, PDGRAPHER accurately predicts its primary targets and suggests additional genes (HTR1A, BRAF, HOXC6), enriching our understanding of its mechanism in modulating GPCR signaling pathways. These findings underscore PDGRAPHER's potential in identifying therapeutic targets and understanding drug actions. More details can be found in Appendix A.10.

Table 1: Model performance across genetic and chemical datasets in test folds containing novel samples

Dataset	Model	Relative position of ground-truth genes in the predicted ranking	Proportion of samples with a partially accurate prediction	Training time (mins)
Genetic-PPI-Lung	Random	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer genes	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer targets	0.51 ± 0.00	0.00 ± 0.00	0
	scGen	-	-	✓3,644.2
	PDGrapher	**0.65 ± 0.06**	**0.02 ± 0.00**	119.09
Genetic-PPI-Breast	Random	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer genes	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer targets	0.51 ± 0.01	0.00 ± 0.00	0
	scGen	-	-	✓1,287.89
	PDGrapher	**0.67 ± 0.11**	**0.01 ± 0.00**	135.07
Chemical-PPI-Lung	Random	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer genes	0.50 ± 0.00	0.02 ± 0.00	0
	Cancer targets	0.56 ± 0.00	0.05 ± 0.00	0
	scGen	-	0.03 ± 0.01	2,018.23
	PDGrapher	**0.83 ± 0.09**	**0.13 ± 0.01**	78.20
Chemical-PPI-Breast	Random	0.50 ± 0.00	0.00 ± 0.00	0
	Cancer genes	0.50 ± 0.00	0.02 ± 0.00	0
	Cancer targets	0.55 ± 0.00	0.05 ± 0.00	0
	scGen	-	0.03 ± 0.00	2,782.65
	PDGrapher	**0.89 ± 0.05**	**0.14 ± 0.02**	118.72

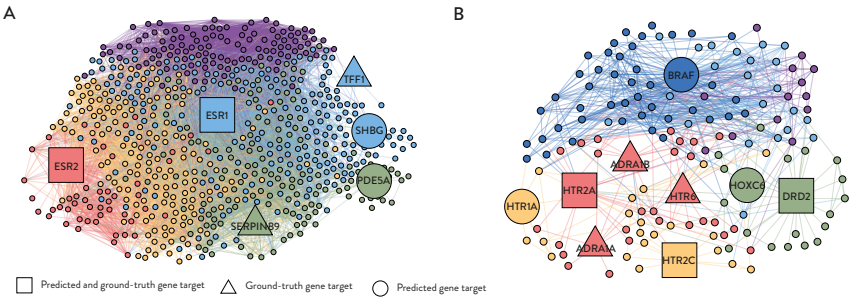

Figure 2: **PDGRAPHER illuminates mode of action of perturbagens. (A,B)** We visualize ground-truth, and predicted therapeutic targets for Raloxifene (A) and Sertindole (B) in Chemical-PPI-Lung using Gephi with ForceAtlas embedding. We highlight in different colors distinct communities identified by Gephi's modularity algorithm.

Ablation study. We perform an ablation study to analyze components in PDGRAPHER's objective function across the chemical datasets. We train PDGRAPHER using only the cycle loss (PDGRAPHER-Cycle), using only the supervision loss (PDGRAPHER-Super), and using both (PDGRAPHER-SuperCycle) in the random splitting setting on Chemical-PPI-Lung and Chemical-PPI-Breast datasets. PDGRAPHER-SuperCycle appears as the best compromise between accuracy in predicting therapeutic genes and reconstruction of treated samples from diseased samples upon intervening on the predicted genes (Figure 6).

5 CONCLUSIONS

We introduce a novel problem formulation for phenotype-driven lead discovery. Given a diseased sample, the goal is to find genes that a genetic or chemical perturbagen should target to shift the sample to a treated state. In practice, this problem translates to predicting a combination of gene targets; therefore, we refer to this formulation as a combinatorial prediction of therapeutic targets. To address this problem, we introduce PDGRAPHER. Given a diseased cell state represented as a gene expression signature and a proxy causal graph of gene-gene interactions, PDGRAPHER predicts candidate target genes to shift the cells to a desired treated state. PDGRAPHER demonstrates superior performance in identifying therapeutic targets across diverse cancer types and intervention datasets. PDGRAPHER's training is also up to 30 times faster than indirect prediction methods like scGen (Lotfollahi et al., 2019) and it can aid in elucidating mechanisms of action of chemical perturbagens, as exemplified in the case of Raloxifene and Sertindole. By flexibly selecting sets of therapeutic targets for intervention, rather than a specific perturbagen, PDGRAPHER enhance the versatility of phenotype-driven lead discovery.

ACKNOWLEDGMENTS

We would like to thank Domen Mohorcic for his help with the PDGRAPHER codebase. We gratefully acknowledge the support of NIH R01-HD108794, US DoD FA8702-15-D-0001, awards from Harvard Data Science Initiative, Amazon Faculty Research, Google Research Scholar Program, AstraZeneca Research, Roche Alliance with Distinguished Scientists, Sanofi iDEA-iTECH Award, Pfizer Research, Chan Zuckerberg Initiative, and Kempner Institute for the Study of Natural and Artificial Intelligence at Harvard University. I.H. was supported, in part, by the Summer Institute in Biomedical Informatics at Harvard Medical School. M.B. and G.G. were supported by the ERC-Consolidator Grant No. 724228 (LEMAN). Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

REFERENCES

Raj Agrawal, Chandler Squires, Karren Yang, Karthikeyan Shanmugam, and Caroline Uhler. ABCD-strategy: Budgeted experimental design for targeted causal structure discovery. *AISTATS 2019 - 22nd International Conference on Artificial Intelligence and Statistics*, 89, 2020.

Sara Aibar, Carmen Bravo González-Blas, Thomas Moerman, Vân Anh Huynh-Thu, Hana Imrichova, Gert Hulselmans, Florian Rambow, Jean Christophe Marine, Pierre Geurts, Jan Aerts, Joost Van Den Oord, Zeynep Kalender Atak, Jasper Wouters, and Stein Aerts. SCENIC: single-cell regulatory network inference and clustering. *Nature Methods 2017 14:11*, 14(11):1083–1086, 10 2017.

Johannes Bange, Esther Zwick, and Axel Ullrich. Molecular targets for breast cancer therapy and prevention. *Nature Medicine*, 7(5):548–552, 2001.

Albert László Barabási, Natali Gulbahce, and Joseph Loscalzo. Network Medicine: A Network-based Approach to Human Disease. *Nature reviews. Genetics*, 12(1):56, 1 2011.

Anastasia Baryshnikova. Systematic Functional Annotation and Visualization of Biological Networks. *Cell Systems*, 2(6):412–421, 2016.

Mathieu Bastian, Sebastien Heymann, and M Jacomy. Gephi: An open source software for exploring and manipulating networks. BT - International AAAI Conference on Weblogs and Social. *International AAAI Conference on Weblogs and Social Media*, pp. 361–362, 2009.

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4): 18–42, 7 2017.

Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. 2021.

Charlotte Bunne, Stefan G Stark, Gabriele Gut, Jacobo Sarabia Del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature Methods*, 20(11):1759–1768, 2023.

Zhong Chen, Ivan S. Yuhanna, Zoya Galcheva-Gargova, Richard H. Karas, Michael E. Mendelsohn, and Philip W. Shaul. Estrogen receptor mediates the nongenomic activation of endothelial nitric oxide synthase by estrogen. *Journal of Clinical Investigation*, 103(3):401–406, 1999.

Fiona Cunningham, James E Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Ruth Bennett, Andrew Berry, Jyothish Bhai, Alexandra Bignell, Konstantinos Billis, Sanjay Boddu, Lucy Brooks, Mehrnaz Charkhchi, Carla Cummins, Luca Da Rin Fioretto, Claire Davidson, Kamalkumar Dodiya, Sarah Donaldson, Bilal El Houdaigui, Tamara El Naboulsi, Reham Fatima, Carlos Garcia Giron, Thiago Genez, Jose Gonzalez Martinez, Cristina Guijarro-Clarke, Arthur Gymer, Matthew Hardy, Zoe Hollis, Thibaut Hourlier, Toby Hunt, Thomas Juettemann, Vinay Kaikala,

Mike Kay, Ilias Lavidas, Tuan Le, Diana Lemos, José Carlos MarugánMarug, Shamika Mohanan, Aleena Mushtaq, Marc Naven, Denye N Ogeh, Anne Parker, Andrew Parton, Malcolm Perry, Ivana Piližota, Irina Prosovetskaia, Manoj Pandian Sakthivel, Ahamed Imran Abdul Salam, Bianca M Schmitt, Helen Schuilenburg, Dan Sheppard, José G Pérez-Silva, William Stark, Emily Steed, Kyösti Sutinen, Ranjit Sukumaran, Dulika Sumathipala, Marie-Marthe Suner, Michal Szpak, Anja Thormann, Francesca Floriana Tricomi, David Urbina-G, Andres Veidenberg, Thomas A Walsh, Brandon Walts, Natalie Willhoft, Andrea Winterbottom, Elizabeth Wass, Marc Chakiachvili, Bethany Flint, Adam Frankish, Stefano Giorgetti, Leanne Haggerty, Sarah E Hunt, Garth R IIsley, Jane E Loveland, Fergal J Martin, Benjamin Moore, Jonathan M Mudge, Matthieu Muffato, Emily Perry, Magali Ruffier, John Tate, David Thybert, Stephen J Trevanion, Sarah Dyer, Peter W Harrison, Kevin L Howe, Andrew D Yates, Daniel R Zerbino, and Paul Flicek. Ensembl 2022. *Database issue Nucleic Acids Research*, 50:989, 2022.

Zizhen Deng, Xiaolong Zheng, H U Tian, and Daniel Dajun Zeng. Deep Causal Learning: Representation, Discovery and Inference. (Ci), 2022.

Robert T. Dorsam and J. Silvio Gutkind. G-protein-coupled receptors and cancer. *Nature Reviews Cancer*, 7(2):79–94, 2007.

Brian J Druker, Shu Tamura, Elisabeth Buchdunger, Sayuri Ohno, Gerald M Segal, Shane Fanning, Jürg Zimmermann, and Nicholas B Lydon. Effects of a selective inhibitor of the abl tyrosine kinase on the growth of bcr–abl positive cells. *Nature Medicine*, 2(5):561–566, 1996.

Erik J.J. Duschek, Louis J. Gooren, and Coen Netelenbos. Effects of raloxifene on gonadotrophins, sex hormones, bone turnover and lipids in healthy elderly men. *European Journal of Endocrinology*, 150(4):539–546, 2004.

Sabri Eyuboglu, Marinka Zitnik, and Jure Leskovec. Mutual interactors as a principle for phenotype discovery in molecular interaction networks. In *Pacific Symposium on Biocomputing*, pp. 61–72, 2023.

Matthias Fey and Jan Eric Lenssen. Fast Graph Representation Learning with PyTorch Geometric. *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 3 2019.

Adam Gayoso, Romain Lopez, Galen Xing, Pierre Boyeau, Valeh Valiollah Pour Amiri, Justin Hong, Katherine Wu, Michael Jayasuriya, Edouard Mehlman, Maxime Langevin, Yining Liu, Jules Samaran, Gabriel Misrachi, Achille Nazaret, Oscar Clivio, Chenling Xu, Tal Ashuach, Mariano Gabitto, Mohammad Lotfollahi, Valentine Svensson, Eduardo da Veiga Beltrame, Vitalii Kleshchevnikov, Carlos Talavera-López, Lior Pachter, Fabian J. Theis, Aaron Streets, Michael I. Jordan, Jeffrey Regier, and Nir Yosef. A Python library for probabilistic analysis of single-cell omics data. *Nature Biotechnology 2022 40:2*, 40(2):163–166, 2 2022.

Robert C Gentleman, Vincent J Carey, Douglas M Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, Kurt Hornik, Torsten Hothorn, Wolfgang Huber, Stefano Iacus, Rafael Irizarry, Friedrich Leisch, Cheng Li, Martin Maechler, Anthony J Rossini, Gunther Sawitzki, Colin Smith, Gordon Smyth, Luke Tierney, Jean YH Yang, and Jianhua Zhang. Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology 2004 5:10*, 5(10):1–16, 9 2004.

Amir Emad Ghassami, Saber Salehkaleybar, Negar Kiyavash, and Elias Bareinboim. Budgeted experiment design for causal structure learning. *35th International Conference on Machine Learning, ICML 2018*, 4:2788–2801, 2018.

Alex Greenfield, Aviv Madar, Harry Ostrer, and Richard Bonneau. DREAM4: Combining Genetic and Dynamic Information to Identify Biological Networks and Dynamical Models. *PLoS ONE*, 5(10), 2010.

Alain Hauser and Peter Bühlmann. Two optimal strategies for active learning of causal models from interventional data. *International Journal of Approximate Reasoning*, 55(4):926–939, 2014. ISSN 0888613X.

Stephen R. Heller, Alan McNaught, Igor Pletnev, Stephen Stein, and Dmitrii Tchekhovskoi. InChI, the IUPAC International Chemical Identifier. *Journal of Cheminformatics*, 7(1):1–34, 5 2015.

Leon Hetzel, Simon Böhm, Niki Kilbertus, Stephan Günnemann, Mohammad Lotfollahi, and Fabian Theis. Predicting Cellular Responses to Novel Drug Perturbations at a Single-Cell Resolution. 2022.

Lukas Heumos, Anna C Schaar, Christopher Lance, Anastasia Litinetskaya, Felix Drost, Luke Zappia, Malte D Lücken, Daniel C Strobl, Juan Henao, Fabiola Curion, Single-Cell Best, Practices Consortium, Herbert B Schiller, and Fabian J Theis. Best practices for single-cell analysis across modalities. *Nature Reviews Genetics 2023*, pp. 1–23, 3 2023.

Brian Hie, Bryan D Bryson, Ellen D Zhong, and Bonnie Berger. Learning Mutational Semantics. (NeurIPS):1–13, 2020.

Vân Anh Huynh-Thu, Alexandre Irrthum, Louis Wehenkel, and Pierre Geurts. Inferring Regulatory Networks from Expression Data Using Tree-Based Methods. *PLOS ONE*, 5(9):e12776, 2010.

Dragomirka Jovic, Xue Liang, Hua Zeng, Lin Lin, Fengping Xu, Yonglun Luo, Yonglun Correspondence, Dragomirka Luo, and Lars Jovic. Single-cell RNA sequencing technologies and applications: A brief overview. *Clinical and Translational Medicine*, 12(3):e694, 3 2022.

Mario F. Juruena, Eduardo Ponde00C9; De Sena, and Irismar Reis De Oliveira. Sertindole in the Management of Schizophrenia. *Journal of Central Nervous System Disease*, 3:JCNSD.S5729, 2011.

Kenji Kamimoto, Blerta Stringa, Christy M Hoffmann, Kunal Jindal, Lilianna Solnica-Krezel, and Samantha A Morris. Dissecting cell identity via network inference and in silico gene perturbation. *Nature*, 614(7949):742–751, 2023.

Michelle M Li, Kexin Huang, and Marinka Zitnik. Graph representation learning in biomedicine and healthcare. *Nature Biomedical Engineering*, 6(12):1353–1369, 2022.

Mohammad Lotfollahi, F. Alexander Wolf, and Fabian J. Theis. scGen predicts single-cell perturbation responses. *Nature Methods 2019 16:8*, 16(8):715–721, 7 2019.

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, Leon Hetzel, Yuge Ji, Ignacio L Ibarra, Sanjay R Srivatsan, Mohsen Naghipourfar, Riza M Daza, Beth Martin, Jay Shendure, Jose L McFaline-Figueroa, Pierre Boyeau, F Alexander Wolf, Nafissa Yakubova, Stephan Günnemann, Cole Trapnell, David Lopez-Paz, and Fabian J Theis. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, pp. e11517, 2023.

Katja Luck, Dae Kyum Kim, Luke Lambourne, Kerstin Spirohn, Bridget E. Begg, Wenting Bian, Ruth Brignall, Tiziana Cafarelli, Francisco J. Campos-Laborie, Benoit Charloteaux, Dongsic Choi, Atina G. Coté, Meaghan Daley, Steven Deimling, Alice Desbuleux, Amélie Dricot, Marinella Gebbia, Madeleine F. Hardy, Nishka Kishore, Jennifer J. Knapp, István A. Kovács, Irma Lemmens, Miles W. Mee, Joseph C. Mellor, Carl Pollis, Carles Pons, Aaron D. Richardson, Sadie Schlabach, Bridget Teeking, Anupama Yadav, Mariana Babor, Dawit Balcha, Omer Basha, Christian Bowman-Colin, Suet Feung Chin, Soon Gang Choi, Claudia Colabella, Georges Coppin, Cassandra D'Amata, David De Ridder, Steffi De Rouck, Miquel Duran-Frigola, Hanane Ennajdaoui, Florian Goebels, Liana Goehring, Anjali Gopal, Ghazal Haddad, Elodie Hatchi, Mohamed Helmy, Yves Jacob, Yoseph Kassa, Serena Landini, Roujia Li, Natascha van Lieshout, Andrew MacWilliams, Dylan Markey, Joseph N. Paulson, Sudharshan Rangarajan, John Rasla, Ashyad Rayhan, Thomas Rolland, Adriana San-Miguel, Yun Shen, Dayag Sheykhkarimli, Gloria M. Sheynkman, Eyal Simonovsky, Murat Taşan, Alexander Tejeda, Vincent Tropepe, Jean Claude Twizere, Yang Wang, Robert J. Weatheritt, Jochen Weile, Yu Xia, Xinping Yang, Esti Yeger-Lotem, Quan Zhong, Patrick Aloy, Gary D. Bader, Javier De Las Rivas, Suzanne Gaudet, Tong Hao, Janusz Rak, Jan Tavernier, David E. Hill, Marc Vidal, Frederick P. Roth, and Michael A. Calderwood. A reference map of the human binary protein interactome. *Nature 2020 580:7803*, 580(7803):402–408, 4 2020.

Malte D. Luecken, M. Büttner, K. Chaichoompu, A. Danese, M. Interlandi, M. F. Mueller, D. C. Strobl, L. Zappia, M. Dugas, M. Colomé-Tatché, and Fabian J. Theis. Benchmarking atlas-level data integration in single-cell genomics. *Nature Methods 2021 19:1*, 19(1):41–50, 12 2021.

Donna Maglott, Jim Ostell, Kim D. Pruitt, and Tatiana Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Research*, 35(Database issue):D26, 1 2007.

Jörg Menche, Amitabh Sharma, Maksim Kitsak, Susan Dina Ghiassian, Marc Vidal, Joseph Loscalzo, and Albert László Barabási. Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347(6224):841, 2 2015. ISSN 10959203.

John G. Moffat, Fabien Vincent, Jonathan A. Lee, Jörg Eder, and Marco Prunotto. Opportunities and challenges in phenotypic drug discovery: An industry perspective. *Nature Reviews Drug Discovery*, 16(8):531–543, 2017.

Jonas Mueller, David N Reshef, George Du, and Tommi Jaakkola. Learning Optimal Interventions. 2016.

Jonas Mueller, David Gifford, and Tommi Jaakkola. Sequence to Better Sequence : Continuous Revision of Combinatorial Structures. (1), 2017.

Aliyu Musa, Laleh Soltan Ghoraie, Shu Dong Zhang, Galina Glazko, Olli Yli-Harja, Matthias Dehmer, Benjamin Haibe-Kains, and Frank Emmert-Streib. A review of connectivity map and computational approaches in pharmacogenomics. *Briefings in Bioinformatics*, 19(3):506–523, 5 2018.

Rose Oughtred, Chris Stark, Bobby Joe Breitkreutz, Jennifer Rust, Lorrie Boucher, Christie Chang, Nadine Kolas, Lara O'Donnell, Genie Leung, Rochelle McAdam, Frederick Zhang, Sonam Dolma, Andrew Willems, Jasmin Coulombe-Huntington, Andrew Chatr-Aryamontri, Kara Dolinski, and Mike Tyers. The BioGRID interaction database: 2019 update. *Nucleic Acids Research*, 47(D1):D529–D541, 1 2019.

Aldo Pacchiano and Robert A Barton. Neural Design for Genetic Perturbation Experiments. pp. 1–37, 2022.

Álvaro Parafita and Jordi Vitrià. Causal Inference with Deep Causal Graphs. 2020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary Devito Facebook, A I Research, Zeming Lin, Alban Desmaison, Luca Antiga, Orobix Srl, and Adam Lerer. Automatic differentiation in PyTorch. Technical report.

Nick Pawlowski, Daniel C. Castro, and Ben Glocker. Deep structural causal models for tractable counterfactual inference. *Advances in Neural Information Processing Systems*, 2020-Decem (NeurIPS), 2020.

Thai Hoang Pham, Yue Qiu, Jucheng Zeng, Lei Xie, and Ping Zhang. A deep learning framework for high-throughput mechanism-driven phenotype compound screening and its application to COVID-19 drug repurposing. *Nature Machine Intelligence 2021 3:3*, 3(3):247–257, 2 2021.

Ruofan Qin, Qingrong Zhao, Chenrui Gu, Chen Wang, Lei Zhang, and Hanguo Zhang. Analysis of oxidase activity and transcriptomic changes related to cutting propagation of hybrid larch. *Scientific Reports*, 13(1):1–12, 2023.

Richard Reindollar, William Koltun, Anna Parsons, Amy Rosen, Suresh Siddhanti, Leo Plouffe, and Beth Israel. Effects of oral raloxifene on serum estradiol levels and other markers of estrogenicity. *Fertility and Sterility*, 78(3):469–472, 2002.

Camilo Ruiz, Marinka Zitnik, and Jure Leskovec. Identification of disease treatment mechanisms through the multiscale interactome. *Nature Communications*, 12(1):1796, 2021.

Deniz Seçilmiş, Thomas Hillerton, Andreas Tjärnberg, Sven Nelander, Torbjörn E.M. Nordling, and Erik L.L. Sonnhammer. Knowledge of the perturbation design is essential for accurate gene regulatory network inference. *Scientific Reports*, 12(1):1–12, 2022.

Qi Song, Matthew Ruffalo, and Ziv Bar-Joseph. Using single cell atlas data to reconstruct regulatory networks. *Nucleic Acids Research*, (34):1–13, 2023. ISSN 0305-1048.

Vasileios Stathias, John Turner, Amar Koleti, Dusica Vidovic, Daniel Cooper, Mehdi Fazel-Najafabadi, Marcin Pilarczyk, Raymond Terryn, Caty Chung, Afoma Umeano, Daniel J.B. Clarke, Alexander Lachmann, John Erol Evangelista, Avi Ma'Ayan, Mario Medvedovic, and Stephan C. Schürer. LINCS Data Portal 2.0: next generation access point for perturbation-response signatures. *Nucleic Acids Research*, 48(D1):D431–D439, 1 2020.

John G. Tate, Sally Bamford, Harry C. Jubb, Zbyslaw Sondka, David M. Beare, Nidhi Bindal, Harry Boutselakis, Charlotte G. Cole, Celestino Creatore, Elisabeth Dawson, Peter Fish, Bhavana Harsha, Charlie Hathaway, Steve C. Jupe, Chai Yin Kok, Kate Noble, Laura Ponting, Christopher C. Ramshaw, Claire E. Rye, Helen E. Speedy, Ray Stefancsik, Sam L. Thompson, Shicai Wang, Sari Ward, Peter J. Campbell, and Simon A. Forbes. COSMIC: the Catalogue Of Somatic Mutations In Cancer. *Nucleic Acids Research*, 47(D1):D941–D947, 1 2019.

Susan Tweedie, Bryony Braschi, Kristian Gray, Tamsin E.M. Jones, Ruth L. Seal, Bethan Yates, and Elspeth A. Bruford. Genenames.org: the HGNC and VGNC resources in 2021. *Nucleic Acids Research*, 49(D1):D939–D946, 1 2021.

Brigitte Uebelhart, François Herrmann, Imre Pavo, Michael W. Draper, and René Rizzoli. Raloxifene treatment is associated with increased serum estradiol and decreased bone remodeling in healthy middle-aged men with low sex hormone levels. *Journal of Bone and Mineral Research*, 19(9):1518–1524, 2004.

Fabien Vincent, Arsenio Nueda, Jonathan Lee, Monica Schenone, Marco Prunotto, and Mark Mercola. Phenotypic drug discovery: recent successes, lessons learned and new directions, 12 2022.

David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, Nazanin Assempour, Ithayavani Iynkkaran, Yifeng Liu, Adam Maciejewski, Nicola Gale, Alex Wilson, Lucy Chin, Ryan Cummings, Diana Le, Allison Pon, Craig Knox, and Michael Wilson. DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Research*, 46(D1):D1074–D1082, 1 2018.

Kevin Xia, Kai-Zhan Lee, Yoshua Bengio, and Elias Bareinboim. The Causal-Neural Connection: Expressiveness, Learnability, and Inference. 2021.

Kevin Xia, Yushu Pan, and Elias Bareinboim. Neural Causal Models for Counterfactual Identification and Estimation. 2:1–57, 2022.

Chen Yan. "phosphodiesterases, 3',5'-cyclic nucleotide (pdes) in gtopdb v.2023.1". *"IUPHAR/BPS Guide to Pharmacology CITE"*, 2023(1), Apr 2023.

Peng Yu Yang, Wei Kang, Ya Wen Pan, Xian Jun Zhao, and Lei Duan. Overexpression of HOXC6 promotes cell proliferation and migration via MAPK signaling and predicts a poor prognosis in glioblastoma. *Cancer Management and Research*, 11:8167–8179, 2019.

Bo Yuan, Ciyue Shen, Augustin Luna, Anil Korkut, Debora S Marks, John Ingraham, and Chris Sander. CellBox: interpretable machine learning for perturbation biology with application to the design of cancer combination therapy. *Cell Systems*, 12(2):128–140, 2021.

Matej Zečević, Devendra Singh Dhami, Petar Veličković, and Kristian Kersting. Relating graph neural networks to structural causal models. *arXiv:2109.04173*, 2021.

Jiaqi Zhang, Chandler Squires, and Caroline Uhler. Matching a Desired Causal State via Shift Interventions. (NeurIPS), 2021.

Jiaqi Zhang, Louis Cammarata, Chandler Squires, Themistoklis P. Sapsis, and Caroline Uhler. Active Learning for Optimal Intervention Design in Causal Models. 2022.

Jie Zhu, Jingxiang Wang, Xin Wang, Mingjing Gao, Bingbing Guo, Miaomiao Gao, Jiarui Liu, Yanqiu Yu, Liang Wang, Weikaixin Kong, Yongpan An, Zurui Liu, Xinpei Sun, Zhuo Huang, Hong Zhou, Ning Zhang, Ruimao Zheng, and Zhengwei Xie. Prediction of drug efficacy from transcriptional profiles with deep learning. *Nature Biotechnology 2021 39:11*, 39(11):1444–1452, 6 2021.

A APPENDIX

A.1 SPATIAL ENRICHMENT ANALYSIS OF PDGRAPHER'S PREDICTED GENES.

We quantify the spatial enrichment for PDGRAPHER 's predicted therapeutic targets using SAFE (Baryshnikova, 2016), a systematic approach that identifies regions that are over-represented for a feature of interest (Figure 1B-D). SAFE requires networks and annotations for each node as an input. We use the PPI network as input and label gene nodes based on PDGRAPHER's predictions: nodes are labeled as 1 if they are predicted to belong to the therapeutic targets set, and 0 otherwise. We compute enrichment analyses for two chemical compounds in the lung cancer cell line A549 test set: Raloxifene and Sertindole. We apply SAFE with the recommended settings: neighborhoods are defined using the short-path weighted layout metric for node distance and neighborhood radius of 0.15, and p-values are computed using the hypergeometric test with multiple testing correction (1,000 iterations). We use the Python implementation of SAFE: https://github.com/baryshnikova-lab/safepy.

A.2 RELATED WORK

Learning optimal interventions. The problem of learning interventions to achieve a desired state has gained interest in recent years. A few recent works formulate this problem as finding optimal interventions to optimize an associated outcome (Mueller et al., 2016; Pacchiano & Barton, 2022; Mueller et al., 2017; Hie et al., 2020). These works offer varied approaches. For example, Mueller et al. (2016) aim to learn an intervention policy defined by a covariate transformation that produces the largest post-intervention improvement with high uncertainty. Pacchiano & Barton (2022) formalize the task as a bandit optimization problem in which each bandit's arm corresponds to a covariate to intervene, and the goal is to recover an almost optimal arm in the least number of arm pulls possible. Mueller et al. (2017) and Hie et al. (2020) approach the problem of sequence-based data where each sequence is associated with an outcome, and the goal is to find mutations in the input sequence that increase a desired outcome. Other recent works formulate this problem as finding optimal interventions to shift the system to a desired state. Zhang et al. (2021; 2022) aimed to find an intervention that applied to a distribution helps match a desired distribution. Specifically, given a distribution P over $\mathbf{X}$ and a desired distribution Q over $\mathbf{X}$, the goal is to find an optimal matching intervention I such that P^I best matches Q under some metric. They address the special case of soft interventions (shift interventions) and use the expectation of distributions as the distance metric.

Neural networks and Structural Causal Models (SCMs). Causal representation learning has been a growing trend in recent years (Deng et al., 2022). It aims to combine the strength of traditional causal learning methods with the robust capabilities of deep learning in the face of large and noisy data. Bottlenecks of traditional causal learning methods include unstructured high-dimensional variables, combinatorial optimization problems, unknown intervention, unobserved confounders, selection bias, and estimation bias (Deng et al., 2022). There are three areas in which deep learning helps to overcome these bottlenecks (Deng et al., 2022). First, in learning causal variables from high-dimensional unstructured data. Second, in learning the causal structure between causal variables, called *causal discovery* within the causal inference literature. And third, in facilitating inference of interventional and counterfactual queries. Within the last branch, a promising approach aims to join SCMs and neural models to facilitate interventional and counterfactual querying. Parafita & Vitrià (2020) put forward the requirements that any DL model should fulfill to approximate causal queries and introduced normalizing causal flows as a specific instantiation. Pawlowski et al. (2020) followed a similar approach to introduce a model capable of computing counterfactual queries. Xia et al. (2021) approached the problem differently, introducing a Neural Causal Model (NCM), a type of SCM with neural networks as structural equations. Together with the NCM, they introduced an algorithm that provably performs identification and inference of interventional queries. A follow-up work extended the NCM framework for identification and inference of counterfactual queries (Xia et al., 2022). The concept of NCMs inspires our work by considering the graph in which we operate as a noisy version of a causal graph and our model operating on the graph as a proxy for the structural equations.

Interventions in Graph Neural Networks (GNNs). GNNs are a type of neural model that falls under the umbrella term of geometric deep learning (Bronstein et al., 2017; 2021; Li et al., 2022). These models use graph-structured data to compute transformed representations useful for down-

stream predictive tasks. Their ability to operate over graphs makes them especially relevant to NCMs. Zečević et al. (2021) explored this connection by introducing interventional GNNs, a GNN in which interventions are represented through mutilations in the input graph, and interventional inference as GNN computations on the mutilated graph. We borrow this concept in our work and extend the representational capabilities of GNNs by assigning learnable embeddings to input nodes.

A.3 NOTATION

A calligraphic letter $\mathcal{X}$ indicates a set, an italic uppercase letter X denotes a graph, uppercase $\mathbf{X}$ denotes a matrix, lowercase $\mathbf{x}$ denotes a vector, and a monospaced letter $\mathtt{X}$ indicates a tuple. Uppercase letter $\mathrm{v}X$ indicates a random variable and lowercase letter $\mathrm{v}x$ indicates its corresponding value; bold uppercase $\mathbf{X}$ denotes a set of random variables, and lowercase letter $\mathbf{x}$ indicates its corresponding values. We denote $P(\mathbf{X})$ as a probability distribution over a set of random variables $\mathbf{X}$ and $P(\mathbf{X} = \mathbf{x})$ as the probability of $\mathbf{X}$ is equal to the value of $\mathbf{x}$ under the distribution $P(\mathbf{X})$. For simplicity, $P(\mathbf{X} = \mathbf{x})$ is abbreviated as $P(\mathbf{x})$.

A.4 COMMENT ON PROBLEM FORMULATION.

In the SCM framework, the conditional probabilities in equation 3 are computed recursively on the graph, each being an expectation over exogenous variables $\mathbf{E}$. Therefore, node states of the previous time point are not necessary. To translate this query into the representation learning realm, we discard the existence of noise variables and directly try to learn a function encoding the transition from an initial state to a desired state. In transitioning the SCM framework to representation learning, we justify omitting explicit noise variables by focusing on learning deterministic transition functions between states. This approach is underpinned by the SCM's ability to abstract away the specifics of exogenous noise through expected outcomes, thus enabling a simplified yet effective representation of causal mechanisms. By concentrating on these transition functions, we capture the essence of causal relationships, ensuring the model's ability to predict outcomes under interventions with reduced complexity and enhanced interpretability.

Relationship between our task and conventional graph prediction tasks. Given that the prediction for each variable in our problem formulation is dependent only on its parents in a graph, GNNs appear especially suited for this problem. We can formulate the query of interest under a graph representation learning paradigm as: Given a graph $G = (\mathcal{V}, \mathcal{E})$, and paired sets of node attributes $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_m\}$ and node labels $\mathcal{Y} = \{\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_m\}$ where each $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$, with $\mathbf{y}_i \in [0, 1]$, we aim at training a neural message passing architecture that given node attributes $\mathbf{X}_i$ predicts the corresponding node labels $\mathbf{Y}_i$. There are, however, some major differences between our problem formulation and the conventional graph prediction tasks, namely, graph and node classification (summarized in Table 2).

In node classification, a single graph G is paired with node attributes $\mathbf{X}$, and the task is to predict the node labels $\mathbf{Y}$. Our formulation differs in that we have m paired sets of node attributes $\mathcal{X}$ and labels $\mathcal{Y}$ instead of a single set, yet they are similar in that there is a single graph in which GNNs operate. In graph classification, a set of graphs $\mathcal{G} = \{G_1, \ldots, G_m\}$ is paired with a set of node attributes $\mathcal{X} = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_m\}$ and the task is to predict a label for each graph $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_m\}$. Here, graphs have a varying structure, and both the topological information and node attributes predict graph labels. In our formulation, a single graph is combined with each node attribute $\mathbf{X}_i$, and the goal is to predict a label for each node, not for the whole graph.

Table 2: Our problem formulation is similar to conventional node and graph classification tasks, albeit some major differences exist.

Task	Number of graphs	Number of node attribute sets	Label dimensions
Graph Classification	m	m	m x 1 (one for each graph)
Node Classification	1	1	1 x n (one for each node)
Ours	1	m	m x n (one for each node of each graph)

A.5 ADDITIONAL METHODOLOGY DETAILS

Response prediction module. Our response prediction module f_r should learn to map pre-perturbagen node values to post-perturbagen node values through learning relationships between connected nodes (equivalent to learning structural equations in SCMs) and propagating the effects of perturbations downstream in the graph (analogous to the recursive nature of query computations in SCMs).

Given a triplet $< \mathbf{x}^h, \mathcal{U}, \mathbf{x}^d >$, we propose a neural model operating on a mutilated graph, $G^{\mathcal{U}}$ where the node attributes are the concatenation of $\mathbf{x}^h$ and $\mathbf{x}'_{\mathcal{U}}$, predicting diseased node values $\mathbf{x}^d$. Each node i has a two-dimensional attribute vector $\mathbf{d}_i = [\mathbf{x}_i^h \,||\, \mathbf{x}'_{\mathcal{U}}]$, where the first element is its gene expression value $\mathbf{x}_i^h$, and the second is a perturbation flag: a binary label indicating whether a perturbation occurs at node i. In practice, we embed each node feature into a high-dimensional continuous space by assigning learnable embeddings to each node based on the value of each input feature dimension. Specifically, for each node, we use the binary perturbation flag to assign a d-dimensional learnable embedding, which is different between nodes but shared across samples for each node. To embed the gene expression value $\mathbf{x}_i^h \in [0, 1]$, we first calculate thresholds using quantiles to assign the gene expression value into one of the B bins. We use the bin index to assign a d-dimensional learnable embedding, which is different between nodes but shared across samples for each node. To increase our model's representation power, we concatenate a d-dimensional positional embedding (d-dimensional vector initialized randomly following a normal distribution). Concatenating these three embeddings results in an input node representation of dimensionality $3.d$

For each node $i \in \mathcal{V}$, an embedding $\mathbf{z}_i$ is computed using a graph neural network operating on the node's neighbors' attributes. The most general formulation of a GNN layer is:

$$\mathbf{h}'_i = \phi\left(\mathbf{h}_i, \bigoplus_{j \in \mathcal{N}^i} \psi(\mathbf{h}_i, \mathbf{h}_j)\right)$$

where $\mathbf{h}'_i$ represents the updated information of node i, and $\mathbf{h}_i$ represents the information of node i in the previous layer, with embedded $\mathbf{d}_i$ being the input to the first layer. ψ is a *MESSAGE* function, $\bigoplus$ an $AGGREGATE$ function (permutation-invariant), and ϕ is an *UPDATE* function. We obtain an embedding $\mathbf{z}_i$ for node i by stacking K GNN layers. Node embedding $\mathbf{z}_i \in \mathbb{R}$ is then passed to a multilayer feed-forward neural network to obtain an estimate of the post-perturbation node values $\mathbf{x}^d$.

Perturbation discovery module. Our perturbagen prediction module f_p should learn the nodes in the graph that should be perturbed to shift node states (attributes) from diseased $\mathbf{x}^d$ to a desired treated state $\mathbf{x}^t$.

Given a triplet $< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, we propose a neural model operating on graph $G^{\mathcal{U}'}$ with node features $\mathbf{x}^d$ and $\mathbf{x}^t$ that predicts a ranking for each node where the top P ranked nodes should be predicted as the nodes in $\mathcal{U}'$. Each node i has a two-dimensional attribute vector: $\mathbf{d}_i = [\mathbf{x}_i^d \,||\, \mathbf{x}_i^t]$. In practice, we represent these binary features in a continuous space using the same approach as described for our response prediction module f_r.

For each node $i \in \mathcal{V}$, an embedding $\mathbf{z}_i$ is computed using a graph neural network operating on the node's neighbors' attributes. We obtain an embedding $\mathbf{z}_i$ for node i by stacking K GNN layers. Node embedding $\mathbf{z}_i \in \mathbb{R}$ is then passed to a multilayer feed-forward neural network to predict a real-valued number for node i.

A.6 DATASETS PREPROCESSING

We compiled and processed six primary data sources and two additional repositories of biological information.

Human protein-protein interaction network. We built a PPI network by aggregating proteins and connections from BIOGRID (Oughtred et al., 2019) (accessed in March 2022), HuRI (Luck et al., 2020), and Menche et al. (Menche et al., 2015). In this graph, nodes represent human proteins, and

edges exist between nodes if there is physical interaction between the proteins. We downloaded a gene ID mapping file from the HUGO Gene Nomenclature Committee. Using this file, we mapped proteins in BIOGRID and Menche et al. (Menche et al., 2015) from Entrez Gene ID (Maglott et al., 2007) to HUGO Gene Nomenclature Committee ID (Tweedie et al., 2021), and proteins in HuRI from Ensembl Gene ID (Cunningham et al., 2022) to HUGO Gene Nomenclature Committee ID (Tweedie et al., 2021). Our final PPI comprises the union of nodes and edges, resulting in a graph with 15,742 nodes and 222,498 undirected edges.

Gene expression data. We downloaded Library of Integrated Network-Based Cellular Signatures (LINCS (Stathias et al., 2020)) level 3 gene expression data from `https://clue.io/releases/data-dashboard` (accessed in February 2022). Level 3 data consists of quantile-normalized samples across each plate and is the lowest level in the LINCS library that can be compared across plates. LINCS contains gene expression measurements for 12,327 genes upon genetic and chemical interventions. There are 387,317 samples upon CRISPR genetic interventions (treated samples), with 5,156 unique knocked-out genes across 27 unique cell lines. There is an average of 17.18 replicates per cell line-knocked-out gene pair. The number of unique genes knocked out in each cell line varies from 1 to 5,114, with an average of 2,042.14 unique genes knocked out per cell line.

Control data for CRISPR interventions, that is, diseased samples, are genetic interventions that either do not contain a gene-specific sequence or whose gene-specific sequence targets a gene not expressed in the human genome. There is a total of 47,781 diseased samples across 50 cell lines. The number of diseased samples for each cell line varies from 1 to 6,890, with an average of 955.62 diseased samples per cell line.

There are 1,313,292 samples upon chemical interventions (treated samples), with 31,234 unique compounds across 229 unique cell lines. There is an average of 7.96 replicates per cell line-compound pair. The number of compounds tested in each cell line varies from 1 to 19,509, with an average of 719.69 unique compounds tested per cell line. Drugs are administered at different doses and measured at varying time points after treatment. On average, there are 2.73 different doses per compound-cell line pair, with a minimum of 1 and a maximum of 26 different doses. On average, gene expression is measured at 1.25 time points per compound-cell line pair, with a minimum of 1 and a maximum of 13 different time points.

Control data for chemical interventions, that is, diseased samples, is treatment with vehicle (dimethyl sulfoxide). There is a total of 76,795 diseased samples across 226 cell lines. The number of diseased samples for each cell line varies from 1 to 7,336, with an average of 339.80 diseased samples per cell line. On average, gene expression of diseased samples is measured at 1.4 time points, with a minimum of 1 and a maximum of 5 different time points.

We filter cell lines to keep those treated with at least 4,000 unique genetic or chemical perturbagens, resulting in 10 selected cell lines for each genetic and chemical dataset. To find healthy cell line counterparts, we extracted all cell lines with the "Unknown" tumor phase in the downloaded LINCS dataset (N=145). Then, we filtered the cell lines by tissue type. To find the exact match to diseased cell lines, we performed a manual literature search to confirm their experimental use as healthy counterparts. We extracted healthy counterparts for three of the ten diseased cell lines: cell line NL20 as the healthy counterpart for A549, cell line MCF10A as the healthy counterpart for MCF7, and cell line RWPE1 as the healthy counterpart for PC3.

Genetic interventions correspond to gene experiment knockouts in which the gene expression of the knocked-out gene after the intervention is zero. Chemical interventions correspond to small molecule treatments, where each molecule targets one or more proteins. Chemical interventions were performed at different dose levels and measured at different time points. We included replicates measured at all time points and doses. For each cell line and condition (healthy, diseased, and treated), we log-normalized level 3 gene expression data. We applied a min-max normalization to transform gene expression values into the range $[0, 1]$ following established practices in the field.

We match genes in LINCS to proteins in our PPI using the HUGO Gene Nomenclature Committee ID (Tweedie et al., 2021), resulting in 10,716 overlapping genes and 151,839 undirected edges. Furthermore, we excluded treated samples from our datasets whose targeted genes were not included in the PPI.

Table 3: Table with several healthy, diseased, and treated samples for lung cancer (A549), breast cancer (MCF7), and prostate cancer (PC3) across genetic and chemical perturbagens.

Dataset type	Cancer type	Sample type	N samples	Category	N perturbagens
Genetic	Lung cancer	healthy	50	vehicle	-
		diseased	4,327	vector	-
		treated	24,255	CRISPR	3,711
	Breast cancer	healthy	113	untreated	-
		diseased	4,852	vector	-
		treated	18,774	CRISPR	3,090
	Prostate cancer	healthy	185	vector	-
		diseased	6,890	vector	-
		treated	21,229	CRISPR	3,710
Chemical	Lung cancer	healthy	50	vehicle	-
		diseased	5,261	vehicle	-
		treated	23,100	compound	1,041
	Breast cancer	healthy	2,675	untreated	-
		diseased	7,336	vehicle	-
		treated	35,421	compound	1,154
	Prostate cancer	healthy	185	vector	-
		diseased	7,202	vehicle	-
		treated	32,555	compound	1,182

We have healthy, diseased, and treated gene expression samples for each cell line treated with several genetic or chemical perturbagens (Table 3). For healthy counterparts, samples with the corresponding treatment ("vector" for genetic perturbagens, and "vehicle" for chemical perturbagens) are not available, therefore, we use the closest possible one (see "Sample category" in Table 3).

Gene regulatory networks. We computed one gene regulatory network (GRN) for each diseased cell line in each condition (genetic and chemical datasets), using the GENIE3 (Huynh-Thu et al., 2010) algorithm on gene expression values of each diseased cell line. We filtered genes in our gene expression dataset (LINCS) to contain only those in the PPI before running the GRN algorithm for consistency between the PPI and GRNs. GENIE3, introduced in 2010, won the Dialogue for Reverse Engineering Assessments and Methods 4 (DREAM4) challenge (Greenfield et al., 2010), which evaluates the success of GRN inference algorithms on benchmarks of simulated data. GENIE3 was introduced in the open source software for bioinformatics Bioconductor (Gentleman et al., 2004) in 2017 and is still used as a gold-standard for GRN generation (Aibar et al., 2017; Seçilmiş et al., 2022; Qin et al., 2023; Song et al., 2023). It is a model based on an ensemble of regression trees and requires as input a matrix of gene expression levels under various conditions. Notably, this expression data is multifactorial. This means that they represent expression levels resulting from a perturbation over a set of genes rather than from a targeted experiment. Multifactorial expression can be obtained as samples from different patients or other biological systems. Therefore, cell line diseased samples are the closest to the ideal input data for GENIE3. GENIE3 produces a directed graph representing gene-gene regulatory interactions. This is achieved by assigning weights to regulatory links and maximizing weights for more significant links. Then, a significance threshold is used to determine which links are substantial enough to be predicted as a regulatory link. We adapted the threshold to generate GRNs with the same network density as our PPI, which was achieved by keeping 303,678 directed edges.

Disease-associated genes. We extracted disease-associated genes from COSMIC (Tate et al., 2019) (Accessed in October 2022) in addition to expert-curated genes available at `https://cancer.sanger.ac.uk/cosmic/curation`. Genes were represented using the HUGO Gene Nomenclature Committee ID. For each cell line in our dataset, we extracted cancer-causing mutations as the list of genes with "Verified" *Mutation verification status* in COSMIC and present in the list of genes curated by experts. Mapping the resulting genes to our list of genes in the PPI resulted in eight disease-associated genes for lung cancer cell line A549, eight disease-associated genes for breast cancer cell line MCF7, and one disease-associated gene for prostate cancer cell line PC3. Therefore, we filtered out the cell line PC3 and proceeded with only MCF7 and A549.

Drug targets. We downloaded drug-related data from DrugBank (Wishart et al., 2018) (accessed in November 2022). We extracted drug names and synonyms, chemical identifiers, drug-gene targets, and all available synonyms for each gene target. We mapped drugs in DrugBank with chemical perturbagens in LINCS by InChI Key (Heller et al., 2015), resulting in 1,522 out of 31,234 unique LINCS compounds mapped to DrugBank with information of at least one target. We mapped drug targets to our PPI network using the HUGO Gene Nomenclature Committee ID, excluding any drug target that was not mapped. Chemical interventions target multiple genes, with a minimum of 1, a maximum of 300, and an average of 2.44 targets per compound.

List of cancer drugs for cancer targets baseline. We extracted the list of cancer drugs by cancer type from NCI (`https://www.cancer.gov/about-cancer/treatment/types/targeted-therapies/approved-drug-list#targeted-therapy-approved-for-breast-cancer`; Accessed in November 2022). We mapped drug names to DrugBank to obtain cancer drug-gene targets. In total, there are 24 drugs associated with breast cancer (cell line MCF7) and 30 drugs associated with lung cancer (cell line A549).

A.7 INTERVENTIONAL DATASETS

Disease intervention data. Disease intervention datasets consist of gene expression measurements of healthy cell lines, disease-associated genes, and gene expression measurements of diseased cell lines. Gene expression samples of healthy and diseased cell lines were retrieved from LINCS (Stathias et al., 2020), and disease-associated genes were retrieved from COSMIC (Tate et al., 2019), as detailed previously. Each dataset $\mathcal{T} = \{\mathtt{T}_1, \ldots, \mathtt{T}_M\}$ is a collection of paired healthy-diseased cell lines where in each sample $\mathtt{T} =< \mathbf{x}^h, \mathcal{U}, \mathbf{x}^d >$, $\mathbf{x}^h$ corresponds to gene expression values of the healthy cell line, set $\mathcal{U}$ is comprised by a randomized subset of disease-associated genes, and $\mathbf{x}^d$ corresponds to gene expression values of diseased cell lines (that is, upon mutations on genes in $\mathcal{U}$). To select the randomized set of disease-associated genes, we first choose at random a proportion $p \in \{0.25, 0.50, 0.75, 1\}$, and then select N disease-associated genes at random where N is the proportion multiplied by the total number of disease-associated genes. Given that more diseased samples are available than healthy samples (see Table 3) when building the triplets, we select a random sample from the set of healthy samples and, therefore, have non-unique healthy samples during training. In total, we built two datasets of disease interventions: one comprised of gene expression of healthy cell line MCF10A, breast cancer mutations, and gene expression of breast cancer cell line MCF7; the second comprised of gene expression of healthy cell line NL20, lung cancer mutations, and gene expression of lung cancer cell line A549. Find more details on data compilation and processing in previous subsections.

Treatment intervention data - genetic. Genetic treatment intervention datasets consist of single-gene knockout experiments using CRISPR / Cas9-mediated gene knockout. Genetic treatment intervention data comprises gene expression measurements of diseased cell lines, single knocked-out genes, and gene expression measurements of treated cell lines. Gene expression samples of diseased and treated cell lines and knocked-out genes were retrieved from LINCS (Stathias et al., 2020). Each dataset $\mathcal{T} = \{\mathtt{T}_1, \ldots, \mathtt{T}_M\}$ is a collection of paired diseased-treated cell lines where in each sample $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, $\mathbf{x}^d$ corresponds to gene expression values of the diseased cell line, set $\mathcal{U}'$ is comprised by the knocked-out gene, and $\mathbf{x}^t$ corresponds to gene expression values of treated cell lines (that is, upon knocking-out the gene in $\mathcal{U}'$). Given that more treated samples are available than diseased samples (see Table 3) when building the triplets, we select a random sample from the set of diseased samples and, therefore, have non-unique diseased samples during training. In total, we built two datasets of treatment interventions: one comprised of gene expression of diseased cell line MCF7, knocked-out genes, and gene expression of treated cell line MCF7; the second comprised of gene expression of diseased cell line A549, knocked-out genes, and gene expression of treated cell line A549. Find more details on data compilation and processing in previous subsections.

Treatment intervention data - chemical. Chemical treatment intervention datasets consist of chemical compound treatment experiments. Chemical treatment intervention data comprises gene expression measurements of diseased cell lines, chemical compound therapeutic targets, and gene expression measurements of treated cell lines. Gene expression samples of diseased and treated cell lines were retrieved from LINCS, and chemical compound targets were retrieved from DrugBank, as detailed previously. Each dataset $\mathcal{T} = \{\mathtt{T}_1, \ldots, \mathtt{T}_M\}$ is a collection of paired diseased-treated

cell lines where in each sample $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, $\mathbf{x}^d$ corresponds to gene expression values of the diseased cell line, set $\mathcal{U}'$ is comprised by the chemical compound targets, and $\mathbf{x}^t$ correspond to gene expression values of treated cell lines (that is, upon treated with the chemical perturbagen targeting genes in $\mathcal{U}'$). Given that more treated samples are available than diseased samples (see Table 3) when building the triplets, we select a random sample from the set of diseased samples and, therefore, have non-unique diseased samples during training. In total, we built two datasets of treatment interventions: one comprised of gene expression of diseased cell line MCF7, chemical compound target genes, and gene expression of treated cell line MCF7; the second comprised of gene expression of diseased cell line A549, chemical compound target genes, and gene expression of treated cell line A549. Find more details on data compilation and processing in previous subsections.

A.8 BASELINES

- **Random baseline:** Given a sample $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, the random baseline returns N random genes as the prediction of genes in $\mathcal{U}'$, where N is the number of genes in $\mathcal{U}'$.
- **Cancer genes:** Given a sample $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, the cancer genes baseline returns the top N genes from an ordered list where the first M genes are disease-associated genes (cancer-driver genes) and the remaining genes are ranked randomly, and where N is the number of genes in $\mathcal{U}'$.
- **Cancer drug targets:** Given a sample $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, the cancer genes baseline returns the top N genes from an ordered list where the first M genes are cancer drug targets and the remaining genes are ranked randomly, and where N is the number of genes in $\mathcal{U}'$.
- **scGen (Lotfollahi et al., 2019):** scGen is a widely-used gold-standard latent variable model for response prediction (Heumos et al., 2023; Gayoso et al., 2022; Jovic et al., 2022; Luecken et al., 2021). Given a set of observed cell type in control and perturbed state, scGen predicts the response of a new cell type to the perturbagen seen in training. To utilize scGen as a baseline, we first fit it to our LINCS gene expression data for each dataset type to predict response to perturbagens, training one model per perturbagen (chemical or genetic). Then, given a sample of paired diseased-treated cell line states, $\mathtt{T} =< \mathbf{x}^d, \mathcal{U}', \mathbf{x}^t >$, we compute the response of cell line with gene expression $\mathbf{x}^d$ to all perturbagens. The predicted perturbagen is that whose predicted response is closest to $\mathbf{x}^t$ in R^2 score.

A.9 EXPERIMENTAL SETUP

- **Random splits:** Our dataset is split randomly into train and test sets to measure our model performance in an IID setting.
- **Leave-cell-line-out splits:** To measure model performance on unseen cell lines, we train our model with random splits on one cell line and test on a new cell line.

Evaluation setup. For all dataset split settings, our model is trained using 5-fold cross-validation, and metrics are reported as the average on the test set. Within each fold, we further split the training set into training and validation sets (8:2) to perform early stopping: we train the model on the training set until the validation loss has not decreased at least 10^{-5} for 15 continuous epochs.

Evaluation metrics. We report average sample-wise R^2 score, and average perturbagen-wise R^2 score to measure performance in the prediction of $\mathbf{x}^t$. The sample-wise R^2 score is computed as the square of Pearson correlation between the predicted sample $\hat{\mathbf{x}}^t \in \mathbb{R}^N$ and real sample $\mathbf{x}^t \in \mathbb{R}^N$. The perturbagen-wise R^2 score is adopted from scGen. It is computed as the square of Pearson correlation of a linear least-squares regression between a set of predicted treated samples $\hat{\mathbf{X}}^t \in \mathbb{R}^{N \times S}$ and a set of real treated samples $\mathbf{X}^t \in \mathbb{R}^{N \times S}$ for the same perturbagen. Higher values indicate better performance in predicting the treated sample $\mathbf{x}^t$ given the diseased sample $\mathbf{x}^d$ and predicted perturbagen.

We also report the average ranking of real therapeutic gene targets in the predicted ordered list of therapeutic targets to measure the ability of our model to rank targets correctly. We normalize the ranking to the range $[0, 1]$ as $1 - ranking/N$ where N is the total number of genes in our dataset. Higher values indicate better performance; that is, the model ranks ground truth therapeutic targets closer to the top of the predicted list. In addition, we report the proportion of test samples for

which the predicted therapeutic targets set has at least one overlapping gene with the ground-truth therapeutic targets set.

Model implementation and training. We implement PDGRAPHER using PyTorch 1.10.1 (Paszke et al.) and the Torch Geometric 2.0.4 Library (Fey & Lenssen, 2019). The implemented architecture yields a neural network with the following hyperparameters: number of GNN layers and number of prediction layers. We set the number of prediction layers to two and performed a grid search over the number of GNN layers (1-3 layers). We train our model using a 5-fold cross-validation strategy and report PDGRAPHER's performance resulting from the best-performing hyperparameter setting.

Network proximity between predicted and ground truth perturbagens. Let $\mathcal{P}$ be the set of predicted therapeutic targets, $\mathcal{R}$ be the set of ground truth therapeutic targets, and $spd(p, r)$ be the shortest-path distance between nodes in P and R. We measure the closest distance between P and R as:

$$d(P, R) = \frac{1}{|R||P|} \sum_{r \in R} \sum_{p \in P} spd(p, r)$$

A.10 PDGRAPHER ILLUMINATES MODE OF ACTION OF CHEMICAL PERTURBAGENS

To demonstrate PDGRAPHER's ability to illuminate the mechanism of action of therapeutic perturbagens, we analyze PDGRAPHER's predictions for Raloxifene (Figure 2A) and Sertindole (Figure 2B) in Chemical-PPI-Lung in the random splitting setting. We visualize ground truth and predicted combinations of therapeutic targets together with their one-hop neighbors in the protein interaction network using Gephi (Bastian et al., 2009) and the ForceAtlas graphical layout. We utilized the modularity algorithm in Gephi to identify distinct communities within the network. The nodes were subsequently colored based on their modularity class to represent these communities visually.

Raloxifene is a second-generation selective estrogen receptor modulator (SERM) with anti-estrogenic impacts on breast and uterine tissues and estrogenic effects on bone, lipid metabolism, and blood clotting (Wishart et al., 2018). It targets a combination of Estrogen Receptor 1 (ESR1), Estrogen Receptor 2 (ESR2), Trefoil Factor 1 (TFF1), and Serpin Family B Member 9 (SERPINB9). PDGRAPHER correctly predicted ESR1, and ESR2. Additionally, PDGRAPHER predicted Sex Hormone Binding Globulin (SHBG) and Phosphodiesterase 5A (PDE5A) genes as combinatorial therapeutic targets. Notably, Raloxifene treatment has been documented to raise SHBG levels in healthy middle-aged and older men (Uebelhart et al., 2004; Reindollar et al., 2002), and post-menopausal women (Duschek et al., 2004). Therefore, PDGRAPHER's prediction of SHBG can be explained due to the strong connection between Raloxifene and downstream effects on SHBG. The prediction of PDE5A by PDGRAPHER is through its functional relationship with estrogen receptors. Estrogen facilitates vasodilation by engaging its receptors, increasing nitric oxide (NO) production. This NO production is pivotal as it stimulates the synthesis of cyclic guanosine monophosphate (cGMP), resulting in the relaxation of smooth muscle cells and the subsequent dilation of blood vessels (Chen et al., 1999). PDE5A plays a crucial role in this mechanism by hydrolyzing cGMP, thereby modulating the vasodilation process to be both controlled and reversible (Yan, 2023). A disruption in this intricate pathway might lead to changes in the expression or functionality of PDE5A. Such alterations potentially explain the observed link between Raloxifene, a selective estrogen receptor modulator, and the modulation of PDE5A activity.

Sertindole is a second-generation antipsychotic to treat schizophrenia. It acts through antagonistic mechanisms against Dopamine D2 Receptor (DRD2), Serotonin receptors HTR2A, HTR2C, and HTR6, and Alpha 1 Adrenergic receptors ADRA1A and ADRA1B (Juruena et al., 2011). PDGRAPHER accurately predicted DRD2, HTR2A, and HTR2C. It additionally predicted serotonin receptor HTR1A, B-Raf Proto-Oncogene (BRAF), and Homeobox C6 (HOXC6) genes. All Sertindole's gene targets are G-protein coupled receptors (GPCRs), as is the predicted target HTR1A. The predictive involvement of BRAF in response to Sertindole's targeting of GPCRs can be explained by its position in the downstream cascade of GPCR signaling pathways. GPCRs influence various intracellular signaling cascades, including the MAPK/Erk signaling pathway of which BRAF is a critical component (Dorsam & Gutkind, 2007). Additionally, HOXC6 has been shown to promote cell proliferation and migration through the activation of the MAPK pathway (Yang et al., 2019). This implies that changes in BRAF activity, potentially induced by altered GPCR signaling due to Sertindole, may modify the behavior of PDGRAPHER-predicted HOXC6 in the downstream pathway.

A.11 SUPPLEMENTARY FIGURES

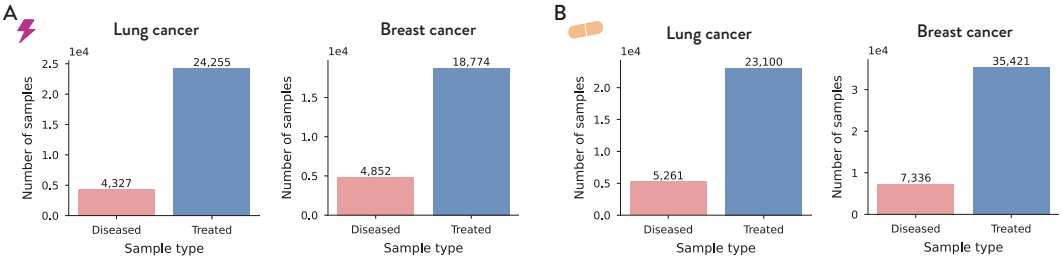

Figure 3: **Overview of interventional datasets. (A,B)** We train PDGRAPHER across two genetic (A) and two chemical (B) intervention datasets, where each dataset is comprised of healthy, diseased, and treated gene expression samples for one cell line (lung cancer cell line A549; breast cancer cell line MCF7), and one proxy causal graph (PPI or GRN). This leads to 8 datasets: Genetic-PPI-Lung, Genetic-PPI-Breast, Chemical-PPI-Lung, Chemical-PPI-Breast, Genetic-GRN-Lung, Genetic-GRN-Breast, Chemical-GRN-Lung, and Chemical-GRN-Breast.

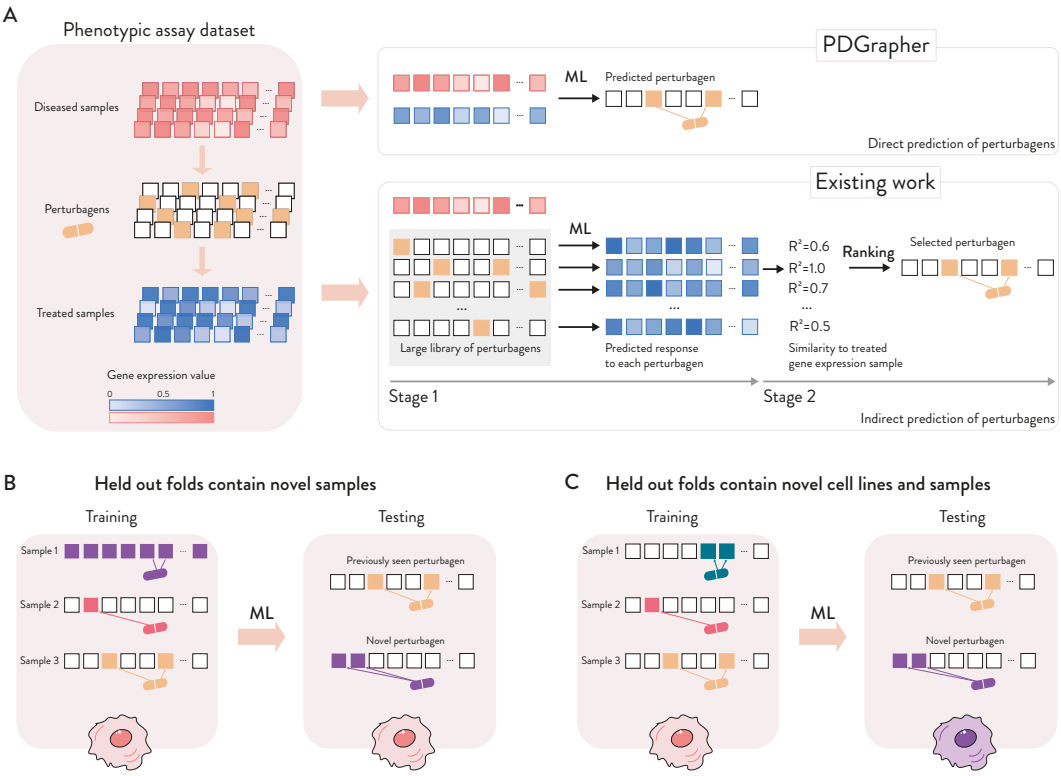

Figure 4: **Overview of evaluation settings and data splits. (A)** Given a dataset with paired diseased and treated samples and a set of perturbagens, PDGRAPHER makes a direct prediction of candidate perturbagens that shift gene expression from a diseased to a treated state, for each disease-treated sample pair. The direct prediction means that PDGRAPHER directly infers the perturbation necessary to achieve a specific response. In contrast to direct prediction of perturbagens, existing methods predict perturbagens only indirectly through a two-stage approach: for a given diseased sample, they learn the response to each one of the perturbagen candidates from an existing library upon intervention and return the perturbagen whose response is as close as possible to the desired treated state. Existing methods learn the response of cells to a given perturbation (Bunne et al., 2023; Lotfollahi et al., 2019; Yuan et al., 2021; Kamimoto et al., 2023), whereas PDGRAPHER focuses on the inverse problem by learning which perturbagen elicit a given response, even in the most challenging cases when the combinatorial composition of perturbagen was never seen before. **(B-C)** We evaluate PDGRAPHER's performance across two settings: randomly splitting samples between training and test set (B), and splitting samples based on the cell line where we train in a cell line and evaluate PDGRAPHER's performance on another cell line the model never encountered before (C).

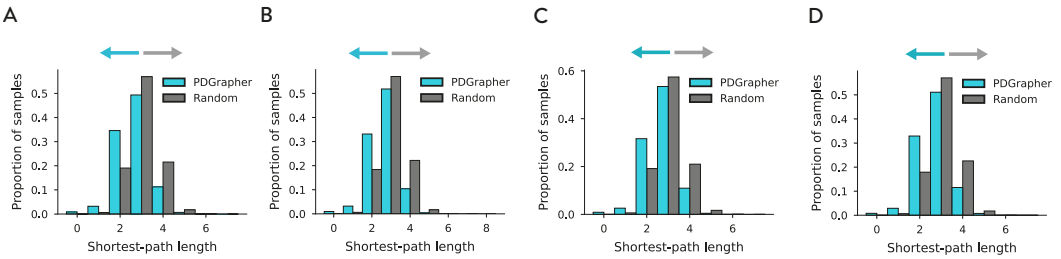

Figure 5: **PDGRAPHER predicts top-ranked genes that are closer to ground-truth therapeutic targets than expected by chance. (A-D)** Sets of therapeutic genes predicted by PDGRAPHER are closer in the network to ground-truth therapeutic genes compared to what would be expected by chance, for Chemical-PPI-Lung (A) and Chemical-PPI-Breast (B) datasets in the random splitting setting and in the leave-cell-out splitting setting (C, D).

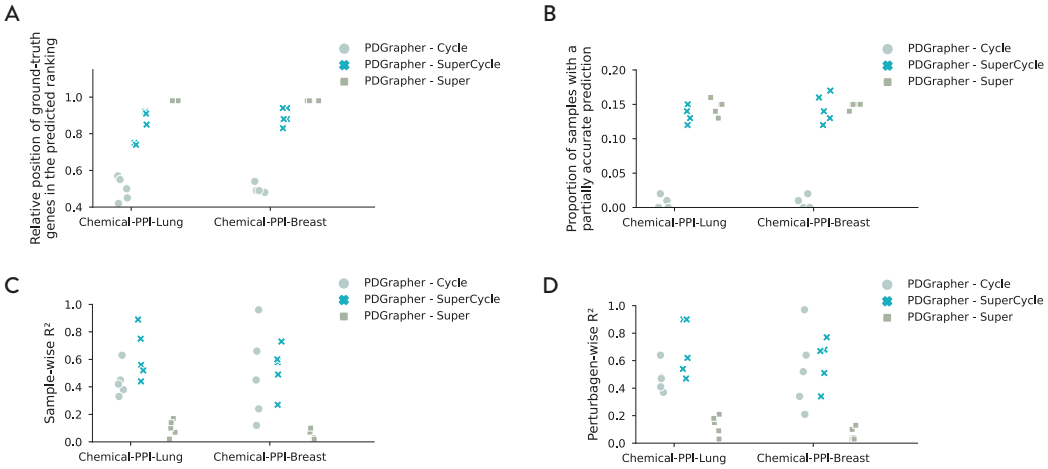

Figure 6: **Ablation study of PDGRAPHER's loss function. (A-D)** Shown are performance metrics of ablation study on PDGRAPHER's objective function components: PDGRAPHER-Cycle trained using only the cycle loss, PDGRAPHER-SuperCycle trained using the supervision and cycle loss, and PDGRAPHER-Super trained using only the supervision loss. Shown is the relative position of ground-truth therapeutic targets in the predicted ranking (A), the proportion of samples with a partially accurate prediction (B), and the sample-wise R^2 (C) and perturbagen-wise R^2 (D) of the reconstruction of treated samples given diseased samples and predicted perturbagens.

Table 4: Model performance across genetic and chemical datasets in test folds containing novel samples and cell lines

Dataset	Model	Relative position of ground-truth genes in the predicted ranking	Proportion of samples with a partially accurate prediction
Genetic-PPI-Lung	Random	0.50 ± 0.00	0.00 ± 0.00
	Cancer genes	0.50 ± 0.00	0.00 ± 0.00
	Cancer targets	0.50 ± 0.00	0.00 ± 0.00
	scGen	-	-
	PDGrapher	**0.64 ± 0.09**	**0.01 ± 0.00**
Genetic-PPI-Breast	Random	0.50 ± 0.00	0.00 ± 0.00
	Cancer genes	0.50 ± 0.00	0.00 ± 0.00
	Cancer targets	0.50 ± 0.00	0.00 ± 0.00
	scGen	-	-
	PDGrapher	**0.65 ± 0.07**	**0.01 ± 0.00**
Chemical-PPI-Lung	Random	0.50 ± 0.00	0.00 ± 0.00
	Cancer genes	0.50 ± 0.00	0.02 ± 0.00
	Cancer targets	0.55 ± 0.00	0.04 ± 0.00
	scGen	-	0.04 ± 0.01
	PDGrapher	**0.90 ± 0.04**	**0.13 ± 0.01**
Chemical-PPI-Breast	Random	0.50 ± 0.00	0.00 ± 0.00
	Cancer genes	0.50 ± 0.00	0.03 ± 0.00
	Cancer targets	0.55 ± 0.00	0.05 ± 0.00
	scGen	-	0.03 ± 0.00
	PDGrapher	**0.82 ± 0.09**	**0.13 ± 0.02**

A

Split type	Dataset	Relative position of ground-truth genes in the predicted ranking	Proportion of samples with a partially accurate prediction
Random	Genetic-GRN-Lung	0.65 ± 0.05	0.01 ± 0.00
	Genetic-GRN-Breast	0.67 ± 0.07	0.01 ± 0.00
	Chemical-GRN-Lung	0.89 ± 0.03	0.14 ± 0.02
	Chemical-GRN-Breast	0.91 ± 0.05	0.12 ± 0.03
Leave cell out	Genetic-GRN-Lung	0.65 ± 0.06	0.01 ± 0.00
	Genetic-GRN-Breast	0.63 ± 0.05	0.01 ± 0.00
	Chemical-GRN-Lung	0.90 ± 0.06	0.10 ± 0.03
	Chemical-GRN-Breast	0.89 ± 0.03	0.13 ± 0.02

Figure 7: **PDGRAPHER predicts genetic and chemical perturbagens to shift cells from diseased to treated states using GRNs. (A)** Shown are performance metrics of PDGRAPHER on Genetic-GRN-Lung, Genetic-GRN-Breast, Chemical-GRN-Lung, and Chemical-GRN-Breast, where we observe similar performance to their counterpart using PPI as the proxy causal graph.