

Predicting transcriptional outcomes of novel multigene perturbations with GEARS

Received: 9 July 2022

Accepted: 12 July 2023

Published online: 17 August 2023

 Check for updatesYusuf Roohani¹ , Kexin Huang² & Jure Leskovec¹  

Understanding cellular responses to genetic perturbation is central to numerous biomedical applications, from identifying genetic interactions involved in cancer to developing methods for regenerative medicine. However, the combinatorial explosion in the number of possible multigene perturbations severely limits experimental interrogation. Here, we present graph-enhanced gene activation and repression simulator (GEARS), a method that integrates deep learning with a knowledge graph of gene–gene relationships to predict transcriptional responses to both single and multigene perturbations using single-cell RNA-sequencing data from perturbational screens. GEARS is able to predict outcomes of perturbing combinations consisting of genes that were never experimentally perturbed. GEARS exhibited 40% higher precision than existing approaches in predicting four distinct genetic interaction subtypes in a combinatorial perturbation screen and identified the strongest interactions twice as well as prior approaches. Overall, GEARS can predict phenotypically distinct effects of multigene perturbations and thus guide the design of perturbational experiments.

The transcriptional response of a cell to genetic perturbation reveals fundamental insights into how the cell functions. Transcriptional responses can describe diverse functionality ranging from how gene regulatory machinery helps maintain cellular identity to how modulating gene expression can reverse disease phenotypes^{1–3}. This has implications for biomedical research, especially in developing personalized therapeutics. For instance, validating drug targets through genetic perturbation studies increases the likelihood of successful clinical trials⁴. Additionally, identifying synergistic gene pairs can enhance the effectiveness of combination therapies^{5–8}. Because complex cellular phenotypes are known to be produced by genetic interactions between small sets of genes, identifying such interactions could facilitate precise cell engineering^{9–14}. While recent advancements have enabled scientists to more rapidly sample perturbation outcomes experimentally^{9,15–19}, computational approaches that predict perturbation effects are indispensable for prioritizing experimental perturbations due to the combinatorial explosion of potential multigene combinations.

However, existing computational methods for predicting perturbational outcomes present their own limitations. The predominant

approach for single-gene perturbation outcome prediction relies on inferring transcriptional relationships between genes in the form of a gene regulatory network^{20–23}. This is limited either by the difficulty in accurately inferring a network from gene expression datasets²⁴ or by the incompleteness of networks derived from public databases^{25–27}. Moreover, existing predictive models built using such networks linearly combine the effects of individual perturbations, which renders them incapable of predicting non-additive effects of multigene perturbations, such as synergy²². More recent work uses deep neural networks trained on data from large perturbational screens to skip the network inference step and directly map genetic relationships into a latent space for perturbation outcome prediction^{28,29}. However, these methods still require that each gene in the combination be experimentally perturbed before the effect of perturbing the combination can be predicted.

Here, we present graph-enhanced gene activation and repression simulator (GEARS), a computational method that integrates deep learning with a knowledge graph of gene–gene relationships to simulate the effects of a genetic perturbation. The incorporation of biological knowledge gives GEARS the ability to predict the outcomes of

¹Department of Biomedical Data Science, Stanford University, Stanford, CA, USA. ²Department of Computer Science, Stanford University, Stanford, CA, USA.

✉ e-mail: jure@cs.stanford.edu

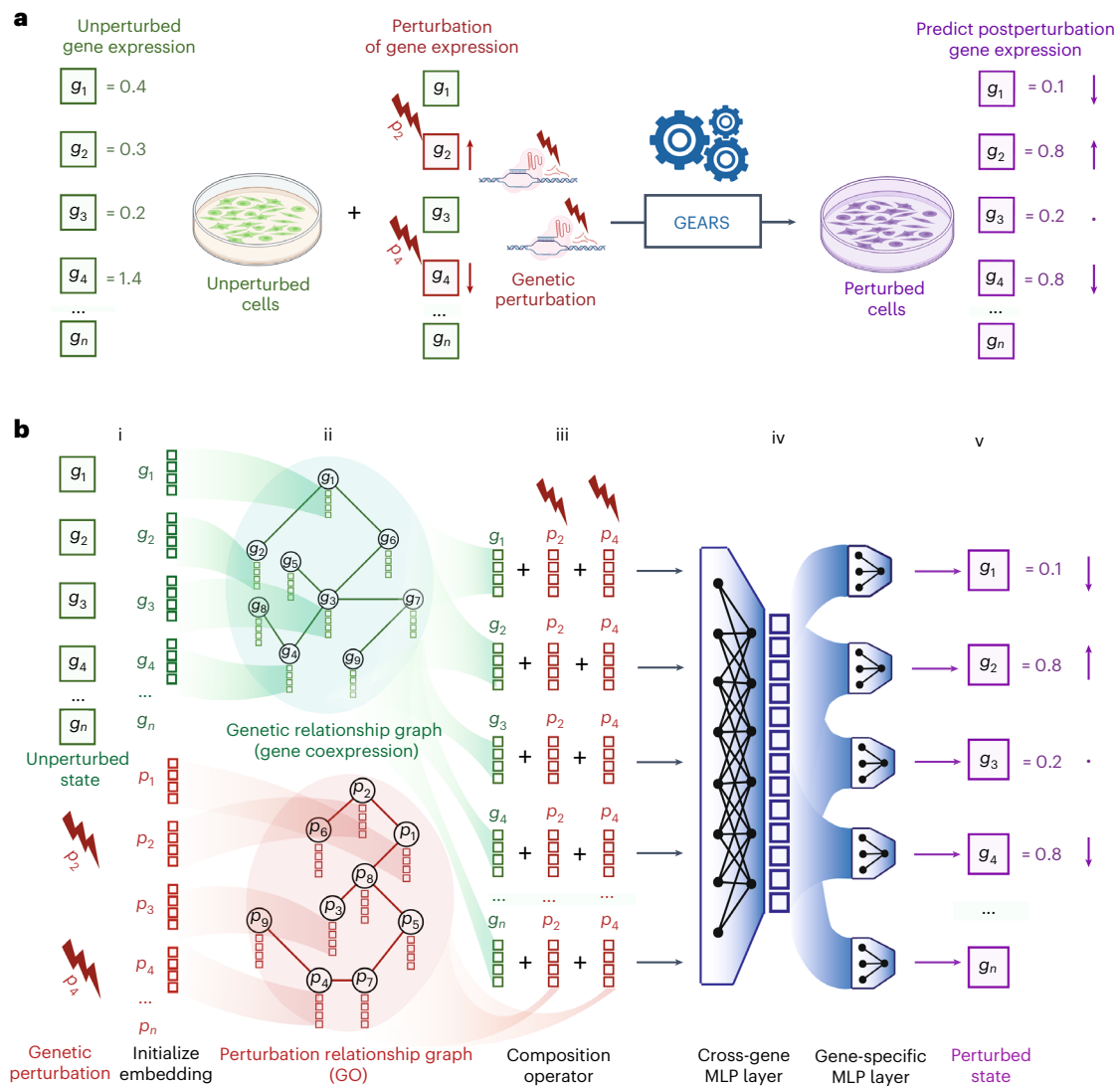


Fig. 1 | GEARs combines prior knowledge with deep learning to predict postperturbation gene expression. **a**, Problem formulation: given unperturbed gene expression (green) and applied perturbation (red), predict the gene expression outcome (purple). Each box corresponds to an individual gene. Arrows indicate change in expression. **b**, GEARs model architecture. (i) For each gene in the unperturbed state, GEARs initializes a gene embedding vector (green) and a gene perturbation embedding vector (red) (ii). These embedding

vectors are assigned as node features in the gene relationship graph and the perturbation relationship graph (iii). A GNN is used to combine information between neighbors in each graph. Each resulting gene embedding is summed with the perturbation embedding of each perturbation in the perturbation set (iv). The output is combined across all genes using the cross-gene layer and fed into gene-specific output layers (v). The final result is postperturbation gene expression; MLP, multilayer perceptron.

perturbing single genes or combinations of genes for which there are no prior experimental perturbation data. GEARs outperformed existing approaches in predicting the outcomes of both one-gene and two-gene perturbations drawn from seven distinct datasets. GEARs could also detect five different genetic interaction subtypes and generalize to new regions of perturbational space by predicting phenotypes that were unlike what was seen during training. Thus, GEARs can directly impact the design of future perturbational experiments.

Results

Knowledge-informed deep learning of perturbation effects

GEARs is a deep learning-based model that predicts the gene expression outcome of combinatorially perturbing a set of one or more genes (perturbation set). Given unperturbed single-cell gene expression along with the perturbation set being applied (Fig. 1a), the output is the transcriptional state of the cell following the perturbation (Methods).

GEARs introduces a new approach of representing each gene and its perturbation using distinct multidimensional embeddings

(arbitrary vectors of numbers used to represent a meaningful concept; Fig. 1b and Supplementary Note 1)^{30,31}. Each gene's embedding is tuned through the course of training to represent key traits of that gene. Splitting the representation into two multidimensional components gives GEARs additional expressivity for capturing gene-specific heterogeneity of perturbation response. Each gene's embedding is sequentially combined with the perturbation embedding of each gene in the perturbation set and finally used to predict the postperturbation state for that gene. This prediction is conditioned on a single 'cross-gene' embedding vector that captures transcriptome-wide information for each cell.

GEARs is uniquely able to predict the outcomes of perturbation sets that involve one or more genes for which there are no experimental perturbation data. GEARs does this by incorporating prior knowledge of gene–gene relationships using a gene coexpression knowledge graph when learning gene embeddings and a Gene Ontology (GO)-derived knowledge graph when learning gene perturbation embeddings (Methods). This relies on two biological intuitions: (i) genes that share similar expression patterns should likely respond

similarly to external perturbations, and (ii) genes that are involved in similar pathways should impact the expression of similar genes after perturbation (Fig. 1b). Different knowledge graphs, such as large context-specific networks, may prove more suitable depending on the gene set of interest³² (Supplementary Note 2). GEARS functionalizes this graph-based inductive bias using a graph neural network (GNN) architecture³³.

Predicting single-gene perturbation transcriptional responses

In the case of single-gene perturbations, GEARS was evaluated on the perturbation of genes whose data had been held out at the time of training, and thus those genes had not been seen experimentally perturbed during training (Fig. 2a). We used data from two different genetic perturbation screens consisting of 1,543 (RPE-1 cells) and 1,092 (K562 cells) perturbations, respectively, with each measuring over 170,000 cells (Replogle et al.³⁴; Supplementary Notes 3 and 4). The screens were run using the Perturb-seq assay, which combines a pooled screen with a single-cell RNA-sequencing readout of the entire transcriptome for each cell¹⁶. GEARS was trained separately on each dataset. In addition to an existing deep learning-based model (CPA), we designed two alternative baseline models for evaluation of performance. One baseline model (no perturbation) assumes that the perturbation does not result in any change in gene expression. The other baseline model first infers a gene regulatory network²⁰ and then linearly propagates the effects of perturbing a gene along this network (adapted from CellOracle²²; Supplementary Notes 6 and 7).

We tested model performance by measuring the mean squared error (m.s.e.; Fig. 2b) and Pearson correlation (Fig. 2c) between the predicted postperturbation gene expression and true postperturbation expression for the held-out set (Supplementary Table 1). Because the vast majority of genes do not show substantial variation between unperturbed and perturbed states, we restricted our m.s.e. analysis to the harder task of only considering the top 20 most differentially expressed genes (Supplementary Note 8). GEARS significantly outperformed all baselines on both datasets with an m.s.e. improvement of 30–50% (Fig. 2b). When considering all genes using Pearson correlation, GEARS exhibited more than two times better performance in the case of both cell lines (Fig. 2c). Additionally, GEARS displayed a clear improvement in capturing the right direction of change in expression following perturbation (Fig. 2d), which reflects a more accurate representation of regulatory relationships. We consistently observed superior performance of GEARS over baselines across metrics (Supplementary Fig. 1) and across five additional datasets, including a genome-wide perturbation screen^{16,18,34–36} (Supplementary Table 2 and Supplementary Figs. 2 and 3). Furthermore, GEARS scaled to large datasets more effectively than conventional gene regulatory network-based methods (Supplementary Table 3). Beyond transcription levels, GEARS also identified groups of genes that induced similar transcriptional responses to perturbation, even when data for their perturbation had not been seen during training (Extended Data Fig. 1 and Supplementary Note 9).

Predicting multigene perturbation outcomes

GEARS is designed to predict transcriptional outcomes for perturbation sets consisting of multiple genes. We evaluated performance using a Perturb-seq dataset (Norman et al.⁹) containing 131 two-gene perturbations. When evaluating GEARS on two-gene perturbations, we defined three generalization classes based on how many of the genes we see experimentally perturbed at the time of training (Fig. 2e). The first case is when the model has seen each of the two genes in the combination individually experimentally perturbed in the training data (two-gene perturbation, zero of two unseen). The other cases, which are progressively harder to predict, are when either one of the two perturbed genes (one of two unseen) or both genes (two of two unseen) have not been

seen individually perturbed at the time of training (Supplementary Fig. 4 and Supplementary Note 10). GEARS improves performance by more than 30% across all cases (Fig. 2f), with the highest improvement of 53% observed when both perturbed genes in the combination are unseen. Improvements were also observed across other metrics (Supplementary Fig. 5) and on a different dataset (Supplementary Tables 2 and 4)³⁷.

Model performance was also analyzed on a gene-by-gene basis. In the case of predicting the outcome of perturbing *FOSB* with *CEBPB*, GEARS correctly captured both the right trend and the magnitude of perturbation across all 20 differentially expressed genes (Fig. 2g) even though one of the perturbed genes (*CEBPB*) had not been seen experimentally perturbed during training. Moreover, the predictions were different from the transcriptional state observed in the case of the single-gene perturbation (*FOSB*) that was seen at the time of training the model (Supplementary Fig. 6). Similar performance was observed for several other examples across generalization categories (Supplementary Fig. 7). We also measured 50% greater enrichment in the most significant differentially expressed genes as predicted by GEARS than observed with baseline methods (Fig. 2h, Extended Data Fig. 2 and Supplementary Note 11).

Although the incorporation of knowledge graphs was instrumental in enabling these predictions (Extended Data Fig. 3 and Supplementary Fig. 8), it also limits the ability of GEARS to predict outcomes for perturbing previously unperturbed genes that are not well connected in this graph (Extended Data Fig. 4 and Supplementary Note 12). GEARS makes use of a Bayesian formulation to overcome this challenge by outputting an uncertainty metric that is inversely correlated with model performance (Supplementary Fig. 9).

Predicting non-additive combinatorial perturbation effects

In the case of a two-gene perturbation, if the outcomes of perturbing the two genes independently are already known, then a naive model could simply add the perturbational effects to estimate the effect of the combinatorial perturbation (Fig. 3a,b). However, genes are known to interact with one another to produce non-additive genetic interactions after perturbation. For example, two genes that independently cause a minor loss in cell growth could synergistically interact with one another following combinatorial perturbation to cause cell death.

We defined five types of genetic interactions (Supplementary Note 15): synergy, suppression, neomorphism, redundancy and epistasis (Supplementary Note 16). When both genes in a two-gene combination had been individually perturbed, the genetic interaction scores predicted by GEARS showed a stronger correlation with the ground truth scores calculated using true expression than existing methods. For instance, the correlation coefficient (R^2) was approximately 0.4 for synergy, neomorphism and redundancy, whereas it was only around 0.0 for the same interactions when predicted by CPA (Extended Data Fig. 5).

To identify new genetic interactions, GEARS can recommend pairs of genes that are predicted to have strong genetic interactions. To assess the real-world application of GEARS where the recommended pairs are then experimentally validated, we calculated performance metrics based on the top-ranked predictions. Precision@10 measures the fraction of predicted combinations in the top ten that truly exhibit a specific genetic interaction subtype, as determined by experimentally measured gene expression after perturbation (Supplementary Note 17). When compared to baseline methods, GEARS improved precision@10 by more than 40% for four of five genetic interaction subtypes, and the improvement exceeded 90% for redundancy and epistasis (Fig. 3c). Additionally, GEARS demonstrated a twofold increase in accuracy when predicting the ten strongest interactions for a specific genetic interaction subtype (top ten accuracy; Extended Data Fig. 6b). Further validation using an additional dataset confirmed the effectiveness of GEARS, showing a 20% increase in accuracy across four genetic interaction subtypes. Moreover, the precision–recall curves for all observed

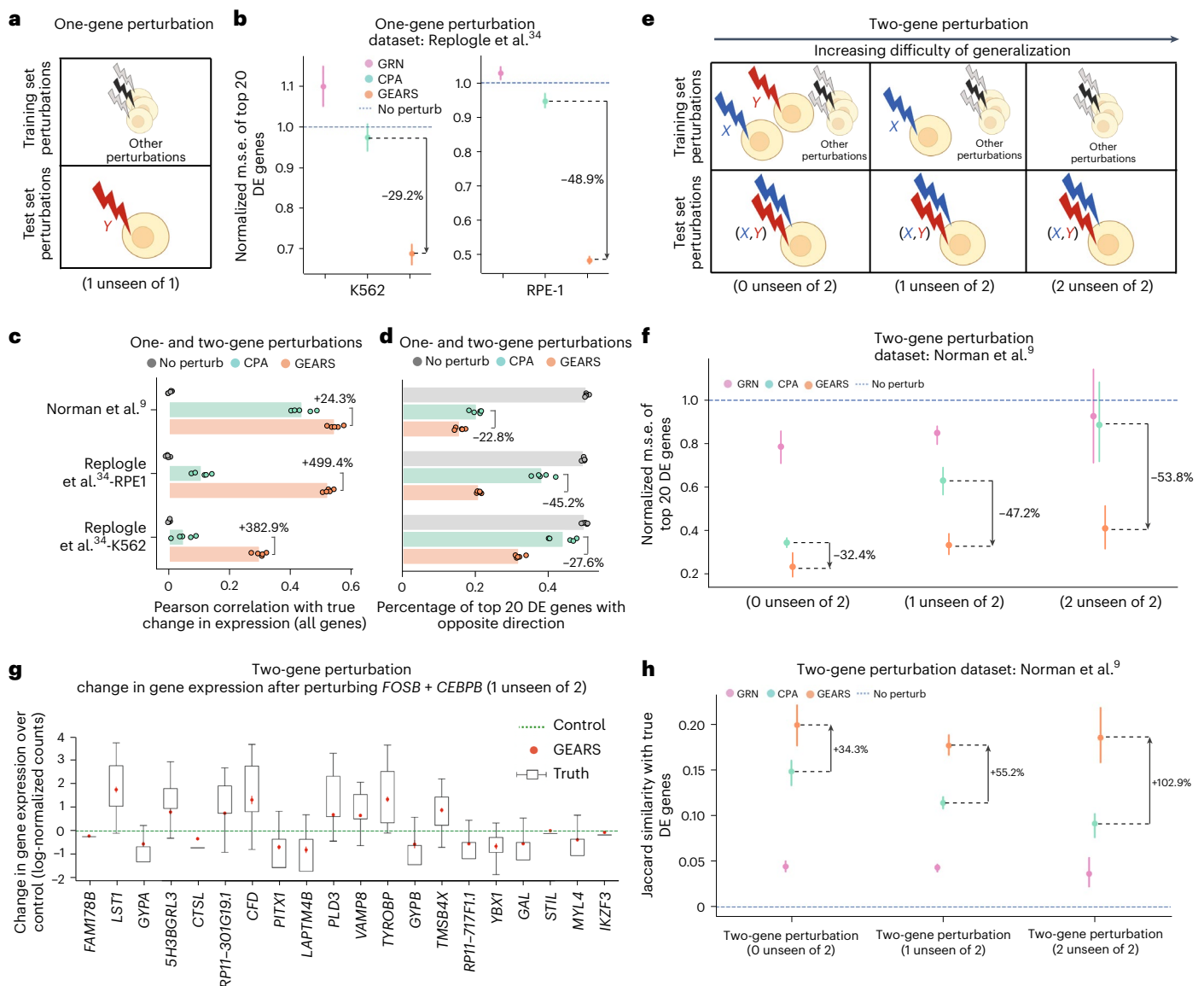


Fig. 2 | GEARs outperforms alternative approaches in predicting postperturbation gene expression. **a**, Train–test data split for single-gene perturbations. **b**, The m.s.e. in predicted postperturbation gene expression for single-gene perturbations normalized to the no perturbation case. For each perturbation, the 20 most differentially expressed (DE) genes were considered; perturb, perturbation; GRN, gene regulatory network. **c**, Pearson correlation between mean predicted postperturbation differential gene expression over control and true values across all genes. **d**, Fraction of the top 20 differentially expressed genes where the predicted postperturbation differential expression is in the opposite direction of the ground truth. **e**, Train–test data split categories for two-gene perturbations. **f**, Normalized m.s.e. in predicted postperturbation

gene expression for two-gene perturbations. **g**, Boxes indicate experimentally measured differential gene expression after perturbing the gene combination *FOSB* and *CEBPB* ($n = 85$). The red symbol shows the mean change in gene expression predicted by GEARs when it has only seen *FOSB* experimentally perturbed at the time of training. The green dotted line shows mean unperturbed control gene expression. Whiskers represent the last data point within 1.5× interquartile range. **h**, Jaccard similarity between model-predicted differentially expressed genes and true differentially expressed genes. Throughout the figure, markers correspond to the mean and error bars correspond to 95% confidence intervals computed over predictions made by five models trained using different data splits ($n = 5$).

genetic interaction subtypes exhibited a higher area under the curve than other methods (Supplementary Fig. 12)³⁷. In scenarios where only one gene had been perturbed previously, GEARs successfully detected synergistic and suppressive interactions (Supplementary Fig. 13).

Different types of genetic interactions can also be evaluated at the level of individual genes. For this, the 20 most affected genes were identified for each two-gene combination (Supplementary Note 18). Based on the m.s.e. for these genes, GEARs was able to capture the effects of different types of genetic interactions more than 40% better than existing methods across three of the five genetic interaction subtypes (Extended Data Fig. 6a). As an example, GEARs predicted the correct

non-additive effects across almost all of the top ten non-additively expressed genes following the perturbation of *PTPN12* and *ZBTB25* (Fig. 3d). This was also observed across other examples belonging to different genetic interaction subtypes (Supplementary Fig. 14).

Predicting new biologically meaningful phenotypes

We applied GEARs to the discovery of new phenotypes by predicting the outcomes of all pairwise combinatorial perturbations of 102 genes from the Norman et al. dataset⁹ (Fig. 4a). To make this prediction, GEARs was trained using the postperturbational gene expression profiles for both one-gene perturbation outcomes and 128 two-gene

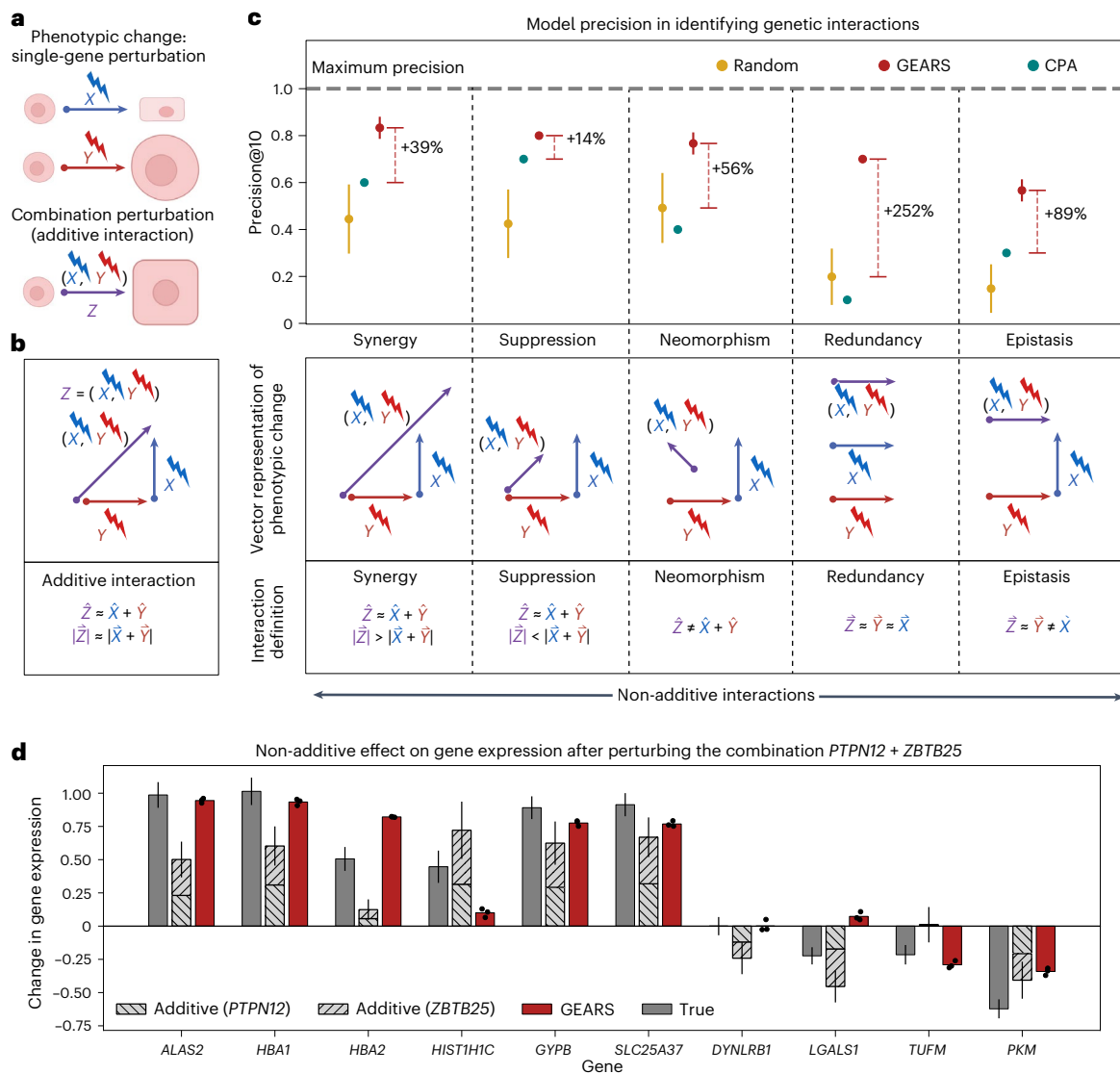


Fig. 3 | GEARS accurately predicts non-additive combinatorial effects and genetic interaction subtypes. a, Illustration of an additive interaction between two genes after perturbation. *X* and *Y* represent change over the unperturbed state caused by single-gene perturbations. *Z* is a combinatorial perturbation of both genes. **b**, Definition of genetic interaction subtypes. **c**, Mean precision@10 in predicting genetic interactions from 131 two-gene combinations (error bars represent s.d.). A random model performs 1,000 random draws; other

models perform three predictions ($n = 3$). **d**, Change in gene expression after perturbing the combination *PTPN12* and *ZBTB25*. The gray bars show the true mean postperturbation gene expression change ($n = 257$). The hatched gray bars show the true change for each of the two single-gene perturbations performed individually (*PTPN12* $n = 164$ and *ZBTB25* $n = 247$), which are summed by the naive additive model. The red bar indicates the prediction made by GEARS ($n = 3$ trained models). Error bars correspond to 95% confidence interval.

perturbation outcomes (Fig. 4b and Supplementary Note 13). The predicted postperturbation expression captured many distinct phenotypic clusters, including those previously identified in Norman et al.⁹ (Fig. 4c and Supplementary Note 13). Additionally, GEARS predicts a few new phenotypes, including one cluster showing high expression of erythroid markers.

To ascertain the biological relevance of this newly predicted phenotype, which was not observed in the training data, we compared it with data for proerythroblasts from the Tabula Sapiens cell atlas (Supplementary Fig. 10 and Supplementary Note 14). While this cluster's distinct high erythroid marker expression has still not been experimentally validated, its identification demonstrates the ability of GEARS to expand the space of postperturbation phenotypes beyond what is observed in perturbational experiments. Moreover, we validated the robustness of this prediction by excluding all phenotypically similar postperturbation outcomes during training (Supplementary Fig. 11).

Mapping combinatorial space of diverse genetic interactions

We extended our analysis to predict genetic interactions among all possible pairwise combinations of 102 genes (Fig. 5a), following CRISPRa-based combinatorial gene activation⁹. By leveraging the predicted postperturbation gene expression for each of the 5,151 pairwise combinatorial perturbations, we constructed a genetic interaction map that could simultaneously represent five distinct types of genetic interactions: synergy, suppression, neomorphism, redundancy and epistasis. The genetic interaction map revealed a rich and diverse landscape of genetic interactions, with many genes exhibiting strong tendencies toward specific genetic interaction subtypes (Fig. 5b). This effect is most evident in the interactions between functionally related genes, which is in line with previous experimental results^{15,16,38}. For instance, genes involved in early erythroid differentiation pathways (*PTPN12*, *IKZF3* and *LHX1*) show a consistent trend of strong synergistic interactions with one another. Moreover, the uniqueness of this genetic interaction map is in how it captures a much broader range of interactions

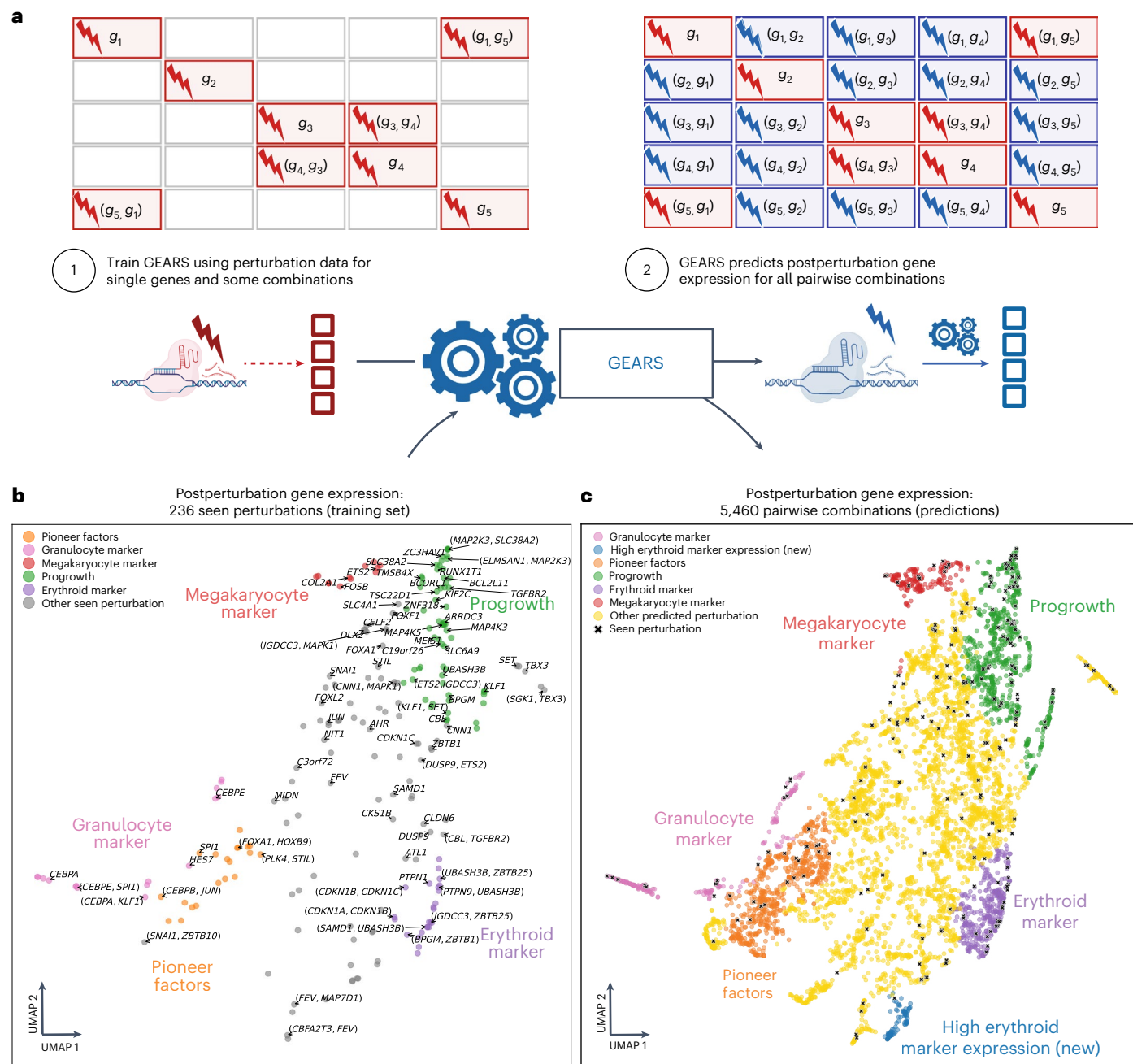


Fig. 4 | GEARs can predict new biologically meaningful phenotypes.

a, Workflow for predicting all pairwise combinatorial perturbation outcomes of a set of genes. **b**, Low-dimensional representation of postperturbation gene expression for 102 one-gene perturbations and 128 two-gene perturbations used to train GEARs. A random selection is labeled. **c**, GEARs predicts

postperturbation gene expression for all 5,151 pairwise combinations of the 102 single genes seen experimentally perturbed. Predicted postperturbation phenotypes (non-black symbols) are often different from phenotypes seen experimentally (black symbols). Colors indicate Leiden clusters labeled using marker gene expression (Supplementary Information).

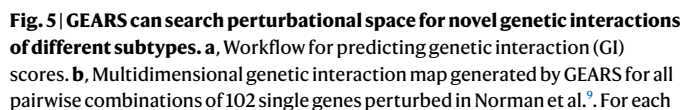
than a conventional genetic interaction map, which focuses primarily on synergistic or buffering interactions (Supplementary Fig. 15)¹⁵.

To validate some of these predictions, we used data from a cell fitness screen that perturbed all pairwise combinations of 92 genes⁹ (Supplementary Note 19). GEARs performed comparably to a real Perturb-seq experiment in capturing the strong interaction effects observed in the cell fitness screen (Extended Data Fig. 7). The distribution of GEARs-predicted genetic interaction scores was significantly higher for perturbations showing synergistic cell fitness effects ($P < 0.0013$, $n = 123$; data were analyzed by one-sided t -test comparing the means) and lower for those showing buffering effects ($P < 4 \times 10^{-5}$, $n = 69$) than those showing approximately additive cell fitness effects.

These findings increase our confidence that several strong interactions captured in the genetic interaction map are biologically meaningful even though not all predictions have been experimentally validated. When trained to directly predict cell fitness, GEARs also showed strong performance (R^2 between 0.64 and 0.93; Supplementary Figs. 16 and 17 and Supplementary Note 20).

Discussion

Recent advancements in high-throughput perturbational screens have enhanced both the precision with which genes can be targeted^{39,40} and the scale of information generated^{17,34}. However, their scalability is limited due to cost. As CRISPR-based perturbational screens become



One of the important strengths of GEARS is detecting emergent interactions between pairs of genes. This feature enhances the discovery of feasible routes for engineering cell identity, where cells are guided between transcriptional states that may be significantly different from one another. For example, GEARS can aid in the precise reengineering of immune cells to prevent exhaustion when targeting cancer^{14,42} or in the reversal of phenotypes linked to aging^{43–45}. Moreover, models like GEARS could predict effective cocktails of transcription factors for reprogramming induced pluripotent stem cells into individual-specific *in vitro* models^{46–50}. Therefore, GEARS holds promise to not only impact the discovery of novel small molecules

for targeting disease but also aid in designing the next generation of cell- and gene-based therapeutics.

Online content

Any methods, additional references, Nature Portfolio reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at <https://doi.org/10.1038/s41587-023-01905-6>.

References

- Kitano, H. Systems biology: a brief overview. *Science* **295**, 1662–1664 (2002).
- Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D. A. & Nolan, G. P. Causal protein-signaling networks derived from multiparameter single-cell data. *Science* **308**, 523–529 (2005).
- Jaitin, D. A. et al. Dissecting immune circuits by linking CRISPR-pooled screens with single-cell RNA-seq. *Cell* **167**, 1883–1896 (2016).
- Nelson, M. R. et al. The support of human genetic evidence for approved drug indications. *Nat. Genet.* **47**, 856–860 (2015).
- Lee, J. S. et al. Synthetic lethality-mediated precision oncology via the tumor transcriptome. *Cell* **184**, 2487–2502 (2021).
- Katti, A., Diaz, B. J., Caragine, C. M., Sanjana, N. E. & Dow, L. E. CRISPR in cancer biology and therapy. *Nat. Rev. Cancer* **22**, 259–279 (2022).
- O'Neil, N. J., Bailey, M. L. & Hieter, P. Synthetic lethality and cancer. *Nat. Rev. Genet.* **18**, 613–623 (2017).
- Haley, B. & Roudnicky, F. Functional genomics for cancer drug target discovery. *Cancer Cell* **38**, 31–43 (2020).
- Norman, T. M. et al. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science* **365**, 786–793 (2019).
- Low, L. A., Mummery, C., Berridge, B. R., Austin, C. P. & Tagle, D. A. Organs-on-chips: into the next decade. *Nat. Rev. Drug Discov.* **20**, 345–361 (2021).
- Wang, H., Yang, Y., Liu, J. & Qian, L. Direct cell reprogramming: approaches, mechanisms and progress. *Nat. Rev. Mol. Cell Biol.* **22**, 410–424 (2021).
- Maude, S. L. et al. Tisagenlecleucel in children and young adults with B-cell lymphoblastic leukemia. *N. Engl. J. Med.* **378**, 439–448 (2018).
- Gillmore, J. D. et al. CRISPR–Cas9 in vivo gene editing for transthyretin amyloidosis. *N. Engl. J. Med.* **385**, 493–502 (2021).
- Lim, W. A. The emerging era of cell engineering: harnessing the modularity of cells to program complex biological function. *Science* **378**, 848–852 (2022).
- Horlbeck, M. A. et al. Mapping the genetic landscape of human cells. *Cell* **174**, 953–967 (2018).
- Dixit, A. et al. Perturb-seq: dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell* **167**, 1853–1866 (2016).
- Frangieh, C. J. et al. Multimodal pooled Perturb-CITE-seq screens in patient models define mechanisms of cancer immune evasion. *Nat. Genet.* **53**, 332–341 (2021).
- Adamson, B. et al. A multiplexed single-cell CRISPR screening platform enables systematic dissection of the unfolded protein response. *Cell* **167**, 1867–1882 (2016).
- Przybyla, L. & Gilbert, L. A. A new era in functional genomics screens. *Nat. Rev. Genet.* **23**, 89–103 (2022).
- Aibar, S. et al. Scenic: single-cell regulatory network inference and clustering. *Nat. Methods* **14**, 1083–1086 (2017).
- Wang, Y., Solus, L., Yang, K. & Uhler, C. Permutation-based causal inference algorithms with interventions. In *Proc. 31st International Conference on Neural Information Processing Systems* (Ed. von Luxburg, U. & Guyon, I.) 5824–5833 (Association for Computing Machinery, 2017).
- Kamimoto, K. et al. Dissecting cell identity via network inference and in silico gene perturbation. *Nature* **614**, 742–751 (2023).
- Friedman, N., Linial, M., Nachman, I. & Pe'er, D. Using Bayesian networks to analyze expression data. *J. Comput. Biol.* **7**, 601–620 (2000).
- Pratapa, A., Jaliha, A. P., Law, J. N., Bharadwaj, A. & Murali, T. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat. Methods* **17**, 147–154 (2020).
- Szklarczyk, D. et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res.* **47**, D607–D613 (2019).
- Kanehisa, M. et al. KEGG for linking genomes to life and the environment. *Nucleic Acids Res.* **36**, D480–D484 (2007).
- Fabregat, A. et al. The reactome pathway knowledgebase. *Nucleic Acids Res.* **46**, D649–D655 (2018).
- Lotfollahi, M., Wolf, F. A. & Theis, F. J. scGen predicts single-cell perturbation responses. *Nat. Methods* **16**, 715–721 (2019).
- Lotfollahi, M. et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Mol. Syst. Biol.* **19**, e11517 (2023).
- Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* **15**, 1053–1058 (2018).
- Eraslan, G., Avsec, Ž., Gagneur, J. & Theis, F. J. Deep learning: new computational modelling techniques for genomics. *Nat. Rev. Genet.* **20**, 389–403 (2019).
- Aytes, A. et al. Cross-species regulatory network analysis identifies a synergistic interaction between *FOXM1* and *CENPF* that drives prostate cancer malignancy. *Cancer Cell* **25**, 638–651 (2014).
- Hamilton, W., Ying, Z. & Leskovec, J. Inductive representation learning on large graphs. In *Proc. 31st International Conference on Neural Information Processing Systems* (Ed. von Luxburg, U. & Guyon, I.) 1025–1035 (Association for Computing Machinery, 2017).
- Replogle, J. M. et al. Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq. *Cell* **185**, 2559–2575 (2022).
- Jost, M. et al. Titrating gene expression using libraries of systematically attenuated CRISPR guide RNAs. *Nat. Biotechnol.* **38**, 355–364 (2020).
- Tian, R. et al. CRISPR interference-based platform for multimodal genetic screens in human iPSC-derived neurons. *Neuron* **104**, 239–255 (2019).
- Replogle, J. M. et al. Combinatorial single-cell CRISPR screens by direct guide RNA capture and targeted sequencing. *Nat. Biotechnol.* **38**, 954–961 (2020).
- Costanzo, M. et al. Global genetic networks and the genotype-to-phenotype relationship. *Cell* **177**, 85–100 (2019).
- Nakamura, M., Gao, Y., Dominguez, A. A. & Qi, L. S. CRISPR technologies for precise epigenome editing. *Nat. Cell Biol.* **23**, 11–22 (2021).
- Hanna, R. E. & Doench, J. G. Design and analysis of CRISPR–Cas experiments. *Nat. Biotechnol.* **38**, 813–823 (2020).
- Bock, C. et al. High-content CRISPR screening. *Nat. Rev. Methods Primers* **2**, 9 (2022).

42. Schmidt, R. et al. CRISPR activation and interference screens decode stimulation responses in primary human T cells. *Science* **375**, eabj4008 (2022).
 43. López-Otín, C., Blasco, M. A., Partridge, L., Serrano, M. & Kroemer, G. Hallmarks of aging: an expanding universe. *Cell* **186**, 243–278 (2023).
 44. Browder, K. C. et al. In vivo partial reprogramming alters age-associated molecular changes during physiological aging in mice. *Nat. Aging* **2**, 243–253 (2022).
 45. Mahmoudi, S., Xu, L. & Brunet, A. Turning back time with emerging rejuvenation strategies. *Nat. Cell Biol.* **21**, 32–43 (2019).
 46. Hendriks, D., Clevers, H. & Artigiani, B. CRISPR–Cas tools and their application in genetic engineering of human stem cells and organoids. *Cell Stem Cell* **27**, 705–731 (2020).
 47. Hsu, M.-N. et al. CRISPR technologies for stem cell engineering and regenerative medicine. *Biotechnol. Adv.* **37**, 107447 (2019).
 48. Ng, A. H. et al. A comprehensive library of human transcription factors for cell fate engineering. *Nat. Biotechnol.* **39**, 510–519 (2021).
 49. Joung, J. et al. A transcription factor atlas of directed differentiation. *Cell* **186**, 209–229 (2023).
 50. Fleck, J. S. et al. Inferring and perturbing cell fate regulomes in human brain organoids. *Nature* <https://doi.org/10.1038/s41586-022-05279-8> (2022).
- Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.
- Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.
- © The Author(s) 2023

Methods

Overview of GEARS

GEARS considers a perturbation dataset of N cells $\mathcal{D} = \{(\mathbf{g}^i, \mathcal{P}^i)\}_{i=1}^N$ where $\mathbf{g}^i \in \mathbb{R}^K$ is the gene expression vector of cell i with K genes, and $\mathcal{P}^i = (P_1^i, \dots, P_M^i)$ is the set of perturbations of size M performed on cell i . $M = 0$ corresponds to an unperturbed cell. Each perturbation P_k in the set corresponds to the index of a gene. The goal of GEARS is to learn a function f that maps a novel perturbation set \mathcal{P} to its postperturbation outcome, which is a gene expression vector \mathbf{g} .

Specifically, given a perturbation set $\mathcal{P} = (P_1, \dots, P_M)$, GEARS first applies a GNN encoder $f_{\text{pert}} : \mathcal{Z} \rightarrow \mathbb{R}^d$ that maps each genetic perturbation $P \in \mathcal{P}$ to a d -dimensional gene perturbation embedding. Another GNN-based encoder $f_{\text{gene}} : \mathcal{Z} \rightarrow \mathbb{R}^d$ maps each gene into a gene embedding. GEARS then combines the set of perturbation embeddings with each of the gene embeddings using a compositional module. A cross-gene decoder $f_{\text{dec}} : \{\mathbb{R}^d\}_{i=1}^K \rightarrow \mathbb{R}^K$ then takes in the set of perturbed gene embeddings and maps them to the postperturbation gene expression vector. The entire network is trained end to end with an autofocus direction-aware loss (Supplementary Note 22).

Gene coexpression graph encoder

To capture the relative heterogeneity of perturbational response for each gene, GEARS represents each gene $u \in \mathcal{Z}$ as a learnable embedding $\mathbf{x}_{\text{gene}}^u \in \mathbb{R}^d$ instead of a scalar. GEARS first obtains a representation for each gene that captures coexpression patterns in the cell. For this, we apply a GNN on a gene coexpression graph $\mathcal{G}_{\text{gene}}$, where edges link coexpressed genes (nodes). GEARS calculates Pearson correlations $\rho_{u,v}$ among genes u, v in the training dataset. For each gene u , we connect it to the top H_{gene} genes that have the highest $\rho_{u,v}$ and are above a threshold δ . Next, we apply a GNN parameterized by θ_g that augments every gene u 's embedding $\mathbf{x}_{\text{gene}}^u$ by integrating information from the embeddings of its coexpressed genes: $\mathbf{h}_{\text{gene}}^u = \text{GNN}_{\theta_g}(\mathbf{x}_{\text{gene}}^u, \mathcal{G}_{\text{gene}}) \in \mathbb{R}^d$.

Incorporating prior knowledge of gene–gene relationships using the GO graph

GEARS predicts the outcome of perturbing genes never seen perturbed before by constructing a gene perturbation similarity graph $\mathcal{G}_{\text{pert}}$, leveraging the pathway information contained in GO⁵¹. We first define \mathcal{G}_{GO} as a bipartite graph where an edge links a gene to a pathway GO term. We denote \mathcal{N}_u as the set of pathways for a gene u . We compute the Jaccard index between a pair of genes u, v as $J_{u,v} = \frac{|\mathcal{N}_u \cap \mathcal{N}_v|}{|\mathcal{N}_u \cup \mathcal{N}_v|}$; this measures the fraction of shared pathways between the two genes. For each gene u , we then select the top H_{pert} gene v with the highest $J_{u,v}$ to construct $\mathcal{G}_{\text{pert}}$. Next, we initialize all possible gene perturbations (P_1, \dots, P_K) with learnable embeddings $(\mathbf{x}_1^{\text{pert}}, \dots, \mathbf{x}_K^{\text{pert}})$. We then feed them into a GNN parameterized by θ_p to augment every perturbation v 's embedding $\mathbf{x}_v^{\text{pert}}$ by integrating information from neighboring perturbations in $\mathcal{G}_{\text{pert}}$: $\mathbf{h}_v^{\text{pert}} = \text{GNN}_{\theta_p}(\mathbf{x}_v^{\text{pert}}, \mathcal{G}_{\text{pert}}) \in \mathbb{R}^d$.

Modeling combinatorial perturbations across genes

Given a perturbation set $\mathcal{P} = (P_1, \dots, P_M)$, GEARS looks up the perturbation embedding of each element of that set $(\mathbf{h}_{P_1}^{\text{pert}}, \dots, \mathbf{h}_{P_M}^{\text{pert}})$. To model multigene perturbations, we use the ‘sum’ compositional operator followed by an MLP: $\mathbf{h}^{\mathcal{P}} = \text{MLP}_{\theta_c}(\sum_{i=1}^M \mathbf{h}_{P_i}^{\text{pert}})$. The ‘sum’ operator allows extendability to perturbations of any size. Thus, each perturbation embedding from $(\mathbf{h}_{P_1}^{\text{pert}}, \dots, \mathbf{h}_{P_M}^{\text{pert}})$ is applied to every gene embedding to obtain a postperturbation gene embedding. For gene u , we have $\mathbf{h}_u^{\text{post-pert}} = \text{MLP}_{\theta_{pp}}(\mathbf{h}_u^{\text{gene}} + \mathbf{h}^{\mathcal{P}})$.

Cross-gene effects and gene-specific decoder

Following application of the perturbations in the embedding space, GEARS maps the postperturbation gene embedding to its corresponding postperturbation gene expression vector. Because each gene has its own perturbation pattern, for every gene u , we apply a gene-specific linear layer parameterized by $\mathbf{w}_u \in \mathbb{R}^d, b_u \in \mathbb{R}$ to map it to a scalar of

perturbation gene expression effect $\mathbf{z}_u = \mathbf{w}_u \mathbf{h}_u^{\text{post-pert}} + b_u \in \mathbb{R}$. We then concatenate the individual effect to a single perturbation effect vector $\mathbf{z} \in \mathbb{R}^K$ for the cell. Because the perturbational effect on a gene can incur secondary effects on other genes, we wanted to use the transcriptome-wide ‘cross-gene’ information for the cell when predicting final gene expression for each gene. Thus, we added an additional MLP that generates a cross-gene embedding for the cell $\mathbf{h}^{\text{cg}} = \text{MLP}_{\theta_{\text{cg}}}(\mathbf{z}) \in \mathbb{R}^d$. Conditioned on this cross-gene state, for every gene u , a gene-specific decoder parameterized by $\mathbf{w}_u^{\text{cg}} \in \mathbb{R}^{d+1}, b_u^{\text{cg}} \in \mathbb{R}$ augments \mathbf{z}_u to $\hat{\mathbf{z}}_u = \mathbf{w}_u^{\text{cg}}(\mathbf{z}_u \parallel \mathbf{h}^{\text{cg}}) + b_u^{\text{cg}} \in \mathbb{R}$, where the double bar notation (\parallel) refers to the vector concatenation operation. Finally, the predicted perturbation effect vector $\hat{\mathbf{z}} \in \mathbb{R}^K$ is added to the gene expression of a randomly sampled unperturbed control cell (\mathbf{g}_{ctrl}) to arrive at the predicted postperturbation gene expression vector for that cell $\hat{\mathbf{g}} = \hat{\mathbf{z}} + \mathbf{g}_{\text{ctrl}}$. This allows GEARS to focus only on learning perturbation effects.

Autofocus direction-aware loss

GEARS optimizes model parameters to fit the predicted $\hat{\mathbf{g}}$ postperturbation gene expression to true postperturbation gene expression \mathbf{g} using stochastic gradient descent. We designed an autofocus loss that automatically gives a higher weight to differentially expressed genes by elevating the exponent of the error. Given a minibatch of T perturbations, where each perturbation k has T_k cells and each cell has K genes with predicted postperturbation gene expression $\hat{\mathbf{g}}$ and true expression \mathbf{g} , the loss is defined as

$$L_{\text{autofocus}} = \frac{1}{T} \sum_{k=1}^T \frac{1}{T_k} \sum_{i=1}^{T_k} \frac{1}{K} \sum_{u=1}^K (\mathbf{g}_u - \hat{\mathbf{g}}_u)^{(2+\gamma)}.$$

However, this loss is insensitive to directionality. To address this, GEARS incorporates an additional direction-aware loss

$$L_{\text{direction}} = \frac{1}{T} \sum_{k=1}^T \frac{1}{T_k} \sum_{i=1}^{T_k} \frac{1}{G} \sum_{u=1}^G [\text{sign}(\mathbf{g}_u - \mathbf{g}_u^{\text{ctrl}}) - \text{sign}(\hat{\mathbf{g}}_u - \mathbf{g}_u^{\text{ctrl}})]^2.$$

The prediction loss function is $L = L_{\text{autofocus}} + \lambda L_{\text{direction}}$, where λ adjusts the weight for the directionality loss.

Uncertainty

GEARS generates an uncertainty score to measure the confidence of model prediction on a novel perturbation. A Gaussian likelihood $\mathcal{N}(\mathbf{g}_u, \hat{\sigma}_u^2)$ is used to model the postperturbation gene expression value for gene u under perturbation \mathcal{P} , where $\hat{\mathbf{g}}_u$ is the predicted postperturbation scalar and $\hat{\sigma}_u^2$ is the variance⁵². We add an additional gene-specific layer to predict the log variance term $s_u = \log \hat{\sigma}_u^2 = \mathbf{w}_u^{\text{unc}} \mathbf{h}_u^{\text{post-pert}} + b_u^{\text{unc}}$ for each gene u and learn it through a modified Bayesian neural network loss⁵²

$$L_{\text{unc}} = \frac{1}{T} \sum_{k=1}^T \frac{1}{T_k} \sum_{i=1}^{T_k} \frac{1}{G} \sum_{u=1}^G \exp(-s_u)(\mathbf{g}_u - \hat{\mathbf{g}}_u)^{(2+\gamma)}.$$

By encouraging log variance to be large when the error is large, the log variance is learned to be a proxy of model uncertainty.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The following are the Gene Expression Omnibus accession numbers used: Dixit et al.¹⁶: GSE90063; Adamson et al.¹⁸: GSE90546; Norman et al.⁹: GSE133344; Jost et al.³⁵: GSE132080; Tian et al.³⁶: GSE124703; Replogle et al.³⁷: GSE146194; Horlbeck et al.¹⁵: GSE116198. The data from Replogle et al.³⁴ are available at <https://doi.org/10.25452/figshare.plus.20022944>.

Code availability

Code to run GEARS is available at <https://github.com/snap-stanford/GEARS>. Results can be reproduced using https://github.com/yhr91/GEARS_misc.

References

51. Consortium, G. O. The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res.* **32**, D258–D261 (2004).
52. Kendall, A. & Gal, Y. What uncertainties do we need in Bayesian deep learning for computer vision? In *Proc. 31st International Conference on Neural Information Processing Systems* (Ed. von Luxburg, U. & Guyon, I.) 5580–5590 (Association for Computing Machinery, 2017).

Acknowledgements

We thank Stephen Quake, Jens Magnusson, Wenfei Sun, Maria Brbic and Hamed Nilforoshan for discussions and for providing feedback on our manuscript. Y.R. acknowledges the support of GlaxoSmithKline. J.L. acknowledges the support of DARPA under Nos. HR00112190039 (TAMI), N660011924033 (MCS); ARO under Nos. W911NF-16-1-0342 (MURI), W911NF-16-1-0171 (DURIP); NSF under Nos. OAC-1835598 (CINES), OAC-1934578 (HDR), CCF-1918940 (Expeditions), NIH under No. 3U54HG010426-04S1 (HuBMAP), Stanford Data Science Initiative, Wu Tsai Neurosciences Institute, Amazon, Docomo, GlaxoSmithKline, Hitachi, Intel, JPMorgan Chase, Juniper Networks, KDDI, NEC, and Toshiba.

Author contributions

Y.R. and J.L. conceived the study. Y.R., K.H. and J.L. performed research, designed the algorithmic framework, analyzed data and wrote the manuscript. J.L. supervised the research.

Competing interests

The authors declare no competing interests.

Additional information

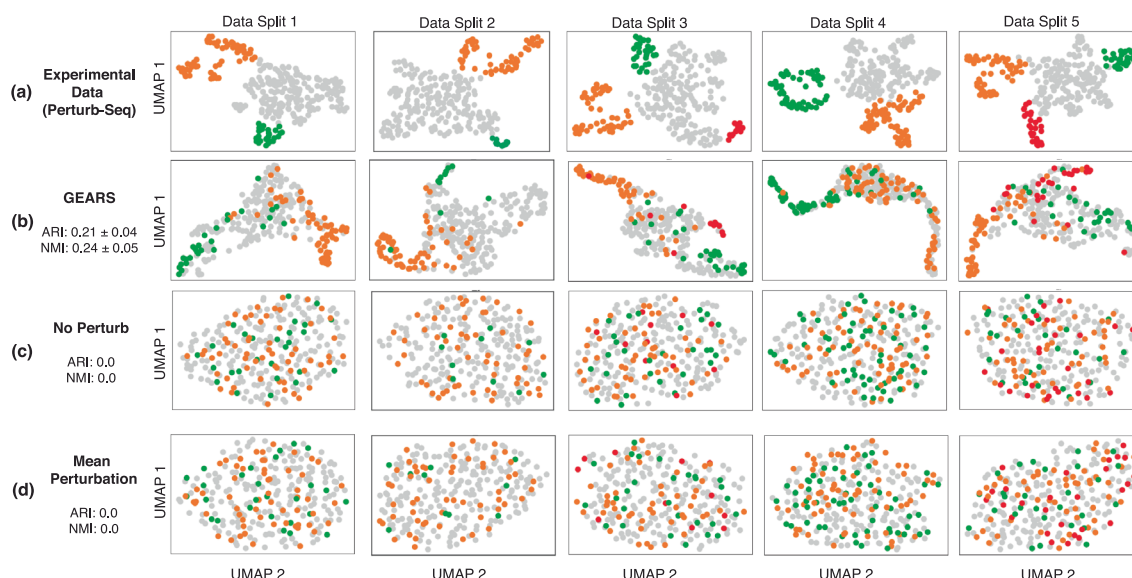
Extended data is available for this paper at <https://doi.org/10.1038/s41587-023-01905-6>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41587-023-01905-6>.

Correspondence and requests for materials should be addressed to Jure Leskovec.

Peer review information *Nature Biotechnology* thanks the anonymous reviewers for their contribution to the peer review of this work.

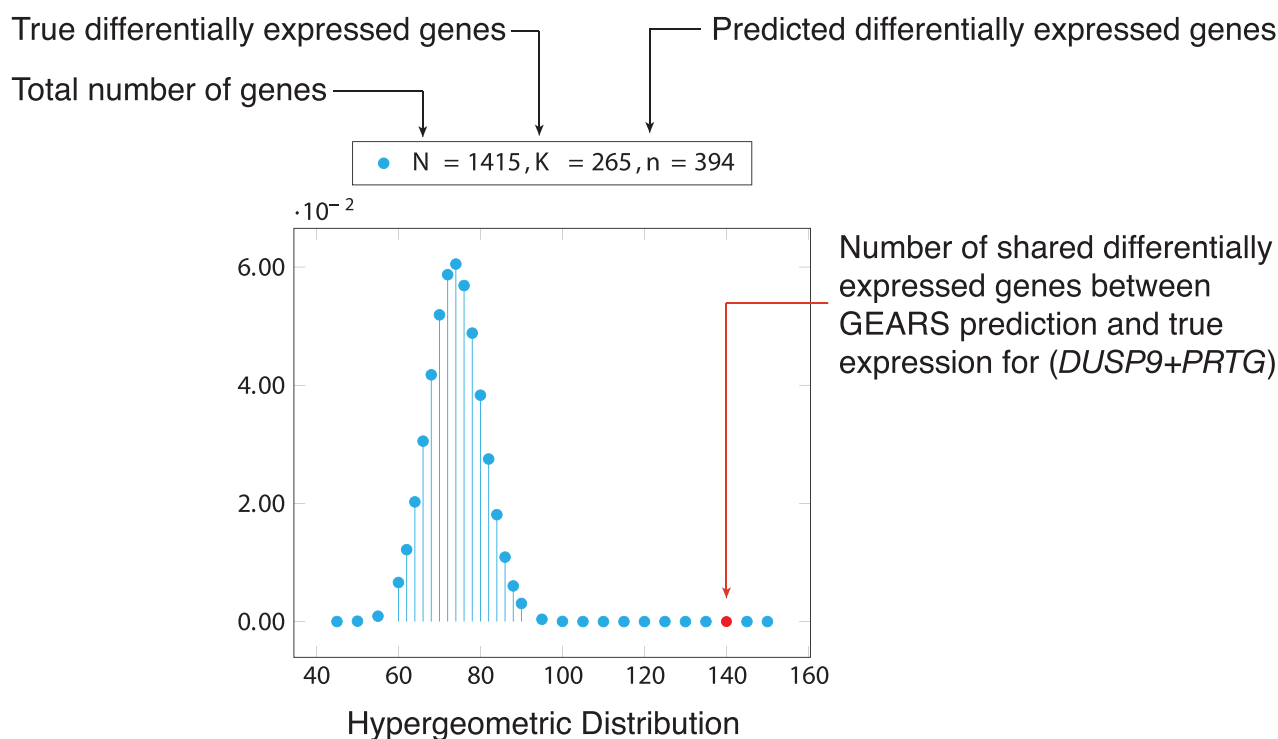
Reprints and permissions information is available at www.nature.com/reprints.



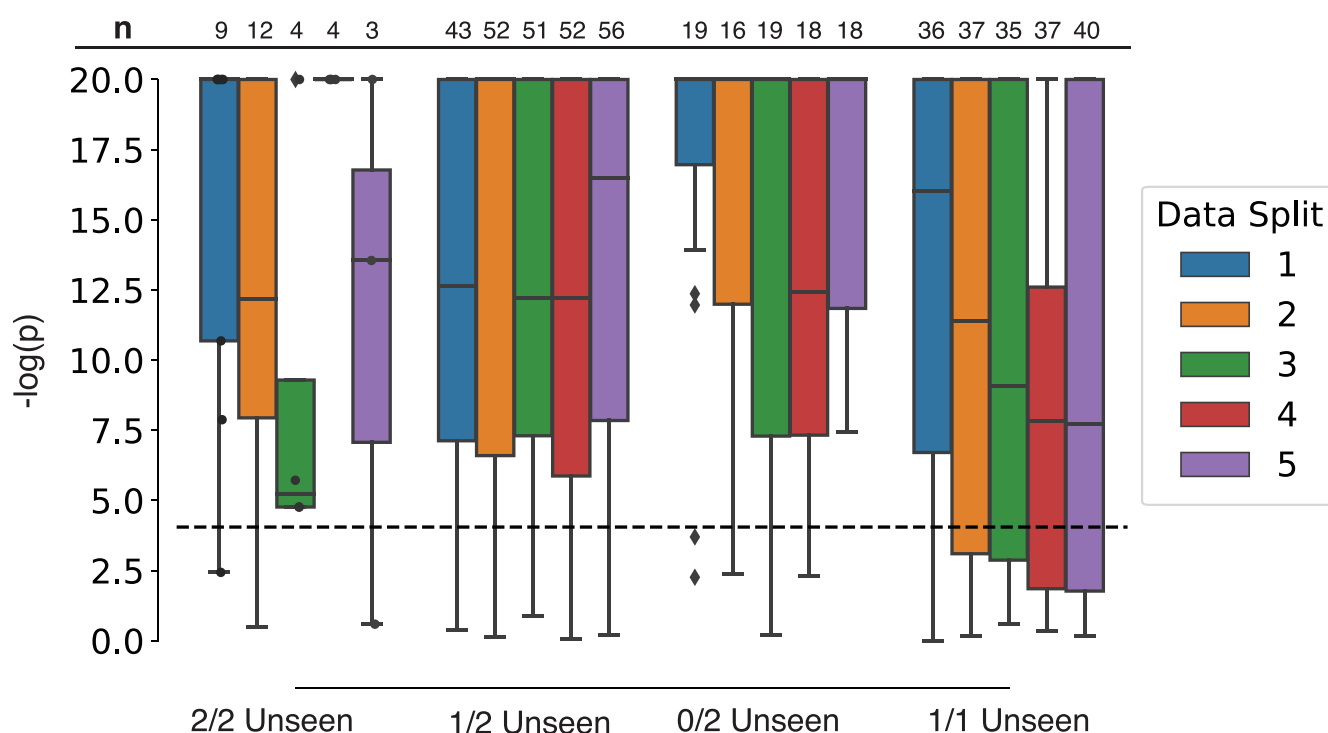
Extended Data Fig. 1 | GEARs identifies groups of genes inducing similar perturbation effect, even when not seen perturbed previously. Each plot presents a low-dimensional (UMAP) representation of postperturbation gene expression following genetic perturbations that were held out in the test set. Each column corresponds to a different split of the experimental data into training and test sets. **a**, Each panel corresponds to true postperturbational transcriptional state measured using a Perturb-Seq assay. Colors correspond to distinct clusters identified using Leiden clustering set to a constant resolution across all panels. The largest cluster is assumed to show minimal perturbation

effect and is colored grey. **b**, Each panel corresponds to postperturbation state predicted by GEARs. Colors correspond to the true labels identified when clustering the true experimental data, thus each point is labeled the same as in **a**. Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) were used to compare clusters identified by GEARs to those observed in true postperturbation expression for each data split. Average values for each metric across splits shown on left. **c**, Same as **b** using a baseline model that predicts no perturbation effect. **d**, Same as **b** using a baseline model that predicts mean perturbation effect.

a Measuring statistical enrichment of true differentially expressed genes in set of differentially expressed genes predicted by GEARS, for a single perturbation



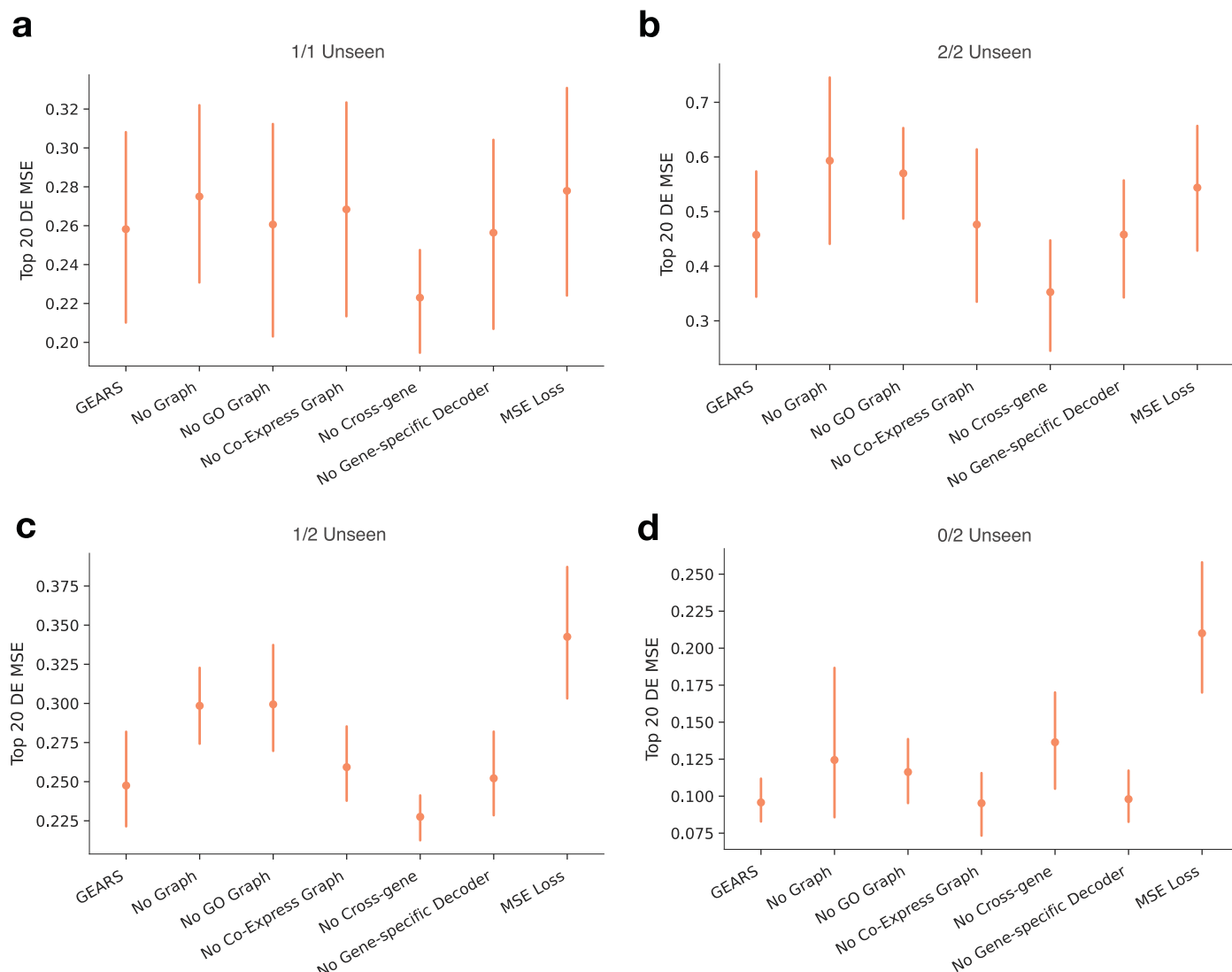
b Statistical significance of enrichment of true differentially expressed genes across all perturbations



Extended Data Fig. 2 | See next page for caption.

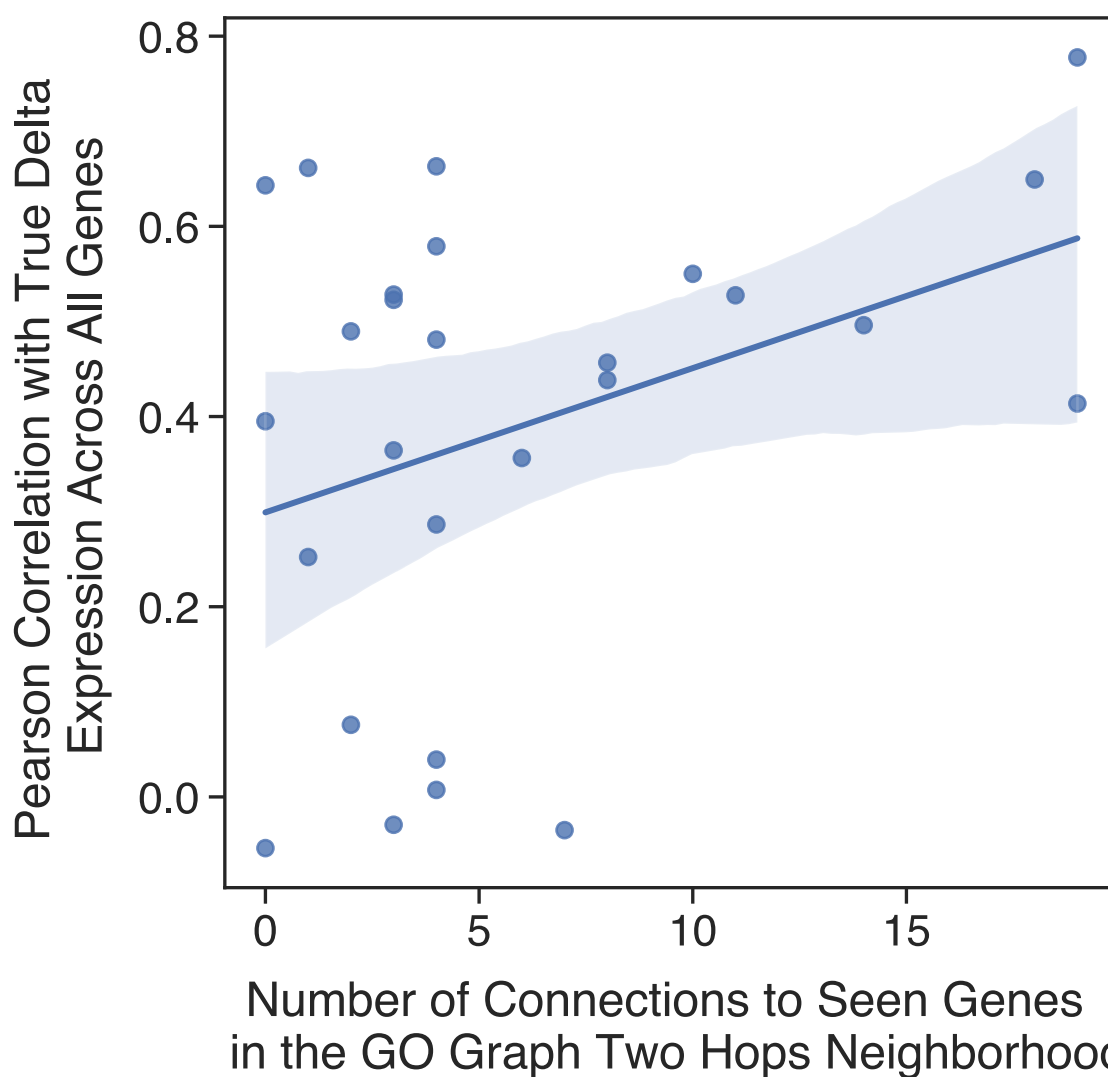
Extended Data Fig. 2 | Identifying significant enrichment for true differentially expressed genes in GEARS predictions. a, Hypergeometric distribution used to model the probability of obtaining a random overlap between the differentially expressed genes predicted by GEARS and the true significantly differentially expressed genes following a perturbation. In this example, 142 genes were shared between GEARS and the true prediction. A p-value is calculated for each perturbation in the held out set. **b,** Box-plot

showing the log (base 10) of the p-value for all held-out perturbations in the Norman et al. 2019 dataset. To account for multiple hypothesis testing (561 tests), a Bonferroni correction was applied, using a significance threshold of 0.05. A black dashed line represents the adjusted threshold. GEARS was trained on 5 different data splits (n=5). Number of data points for each bar are listed above it. Whiskers represent last data point within 1.5x interquartile range below the first quantile and above the third quantile.



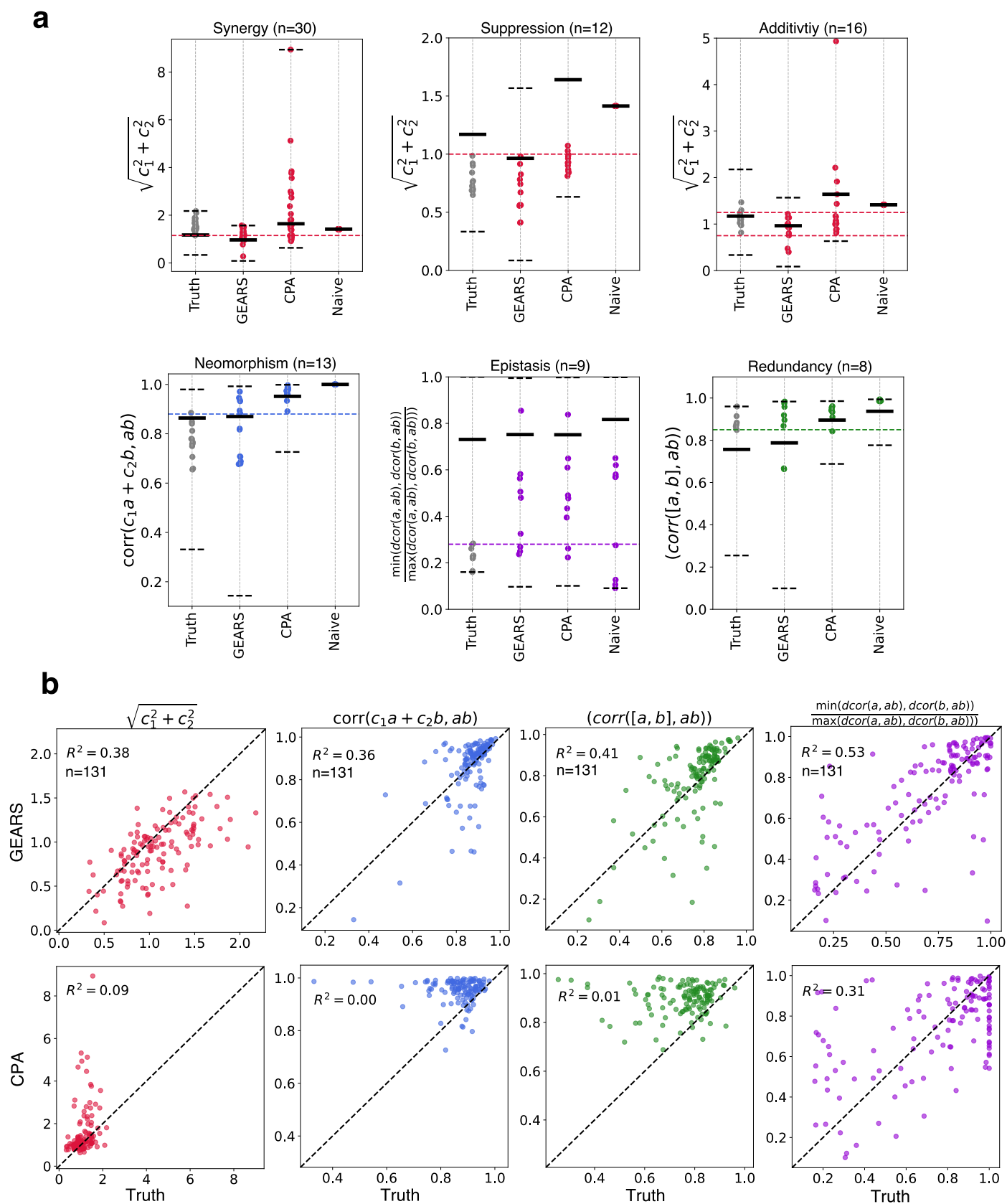
Extended Data Fig. 3 | Model ablation study highlights relative importance of GEARS components under different generalization conditions. The ‘No Graph’ condition removes both the gene ontology graph and co-expression graph; ‘No GO Graph’ removes the gene ontology graph; ‘No Co-Express Graph’ removes the co-expression graph; ‘No Cross-gene’ removes the cross-gene MLP layer; ‘No Gene-specific Decoder’ removes the gene specific decoder MLP and uses a shared MLP instead; ‘MSE Loss’ switches from the auto-focus loss to the regular L2 loss. Four generalization conditions are considered: **a**, (1/1 Unseen) single-gene perturbations not seen experimentally perturbed at the time of

training. **b–d**, (2/2 Unseen) two-gene perturbations in which both genes were not seen experimentally perturbed individually at the time of training (**b**), (1/2 Unseen) one of the two genes was not seen experimentally perturbed (**c**) or (0/2 Unseen) both genes have been seen experimentally perturbed (**d**). Performance is measured using the mean squared error in predicted postperturbation gene expression for the top 20 most differentially expressed genes. For all panels (**a–d**) the marker indicates the mean MSE over predictions made by models trained using 5 different training data splits ($n=5$). The error bars represent bootstrapped 95% CI.



Extended Data Fig. 4 | Model performance relationship with network connectivity. Each point in the scatter plot corresponds to a prediction made for a novel single-gene perturbation not seen at the time of training. The y-axis plots the Pearson correlation between the true mean postperturbation differential

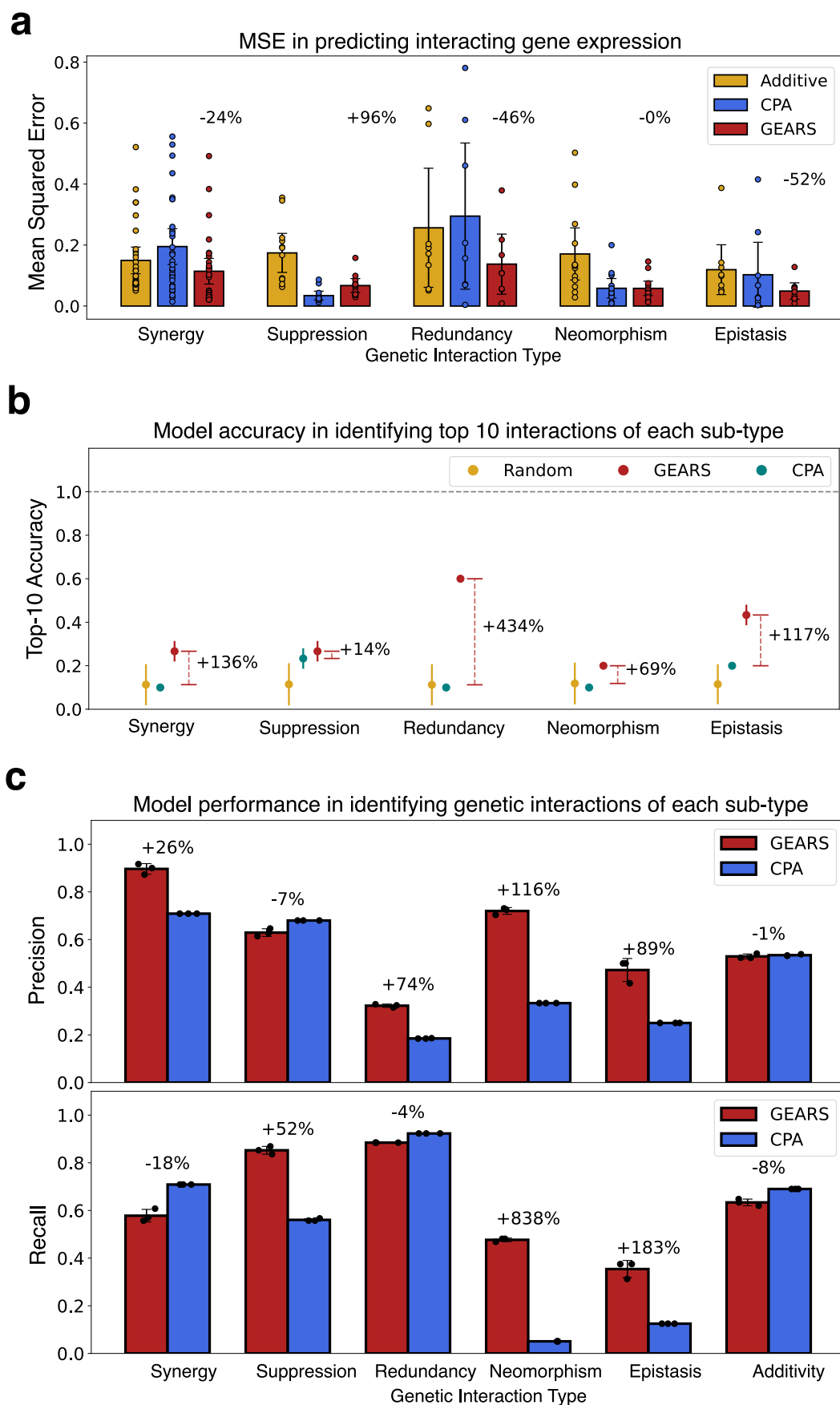
expression over unperturbed control and the same predicted by GEARS. The x-axis measures the number of connections between the novel perturbed gene and other genes in the network that had been seen at the time of training. Error band corresponds to 95% CI.



Extended Data Fig. 5 | See next page for caption.

Extended Data Fig. 5 | Model performance at predicting genetic interaction (GI) scores. a, GI scores for the set of combinatorial perturbations that were defined as expressing a specific GI subtype phenotype in Norman et al. 2019. The gray dots correspond to GI scores computed using true postperturbation gene expression. The colored dots were computed using predicted postperturbation gene expression under three different models: GEARS, CPA and Naive models. The naive model here simply sums together the effects of single-gene perturbations. The metrics on the y-axis correspond to different GI scores and the colored dotted lines indicate the defined thresholds for determining if a

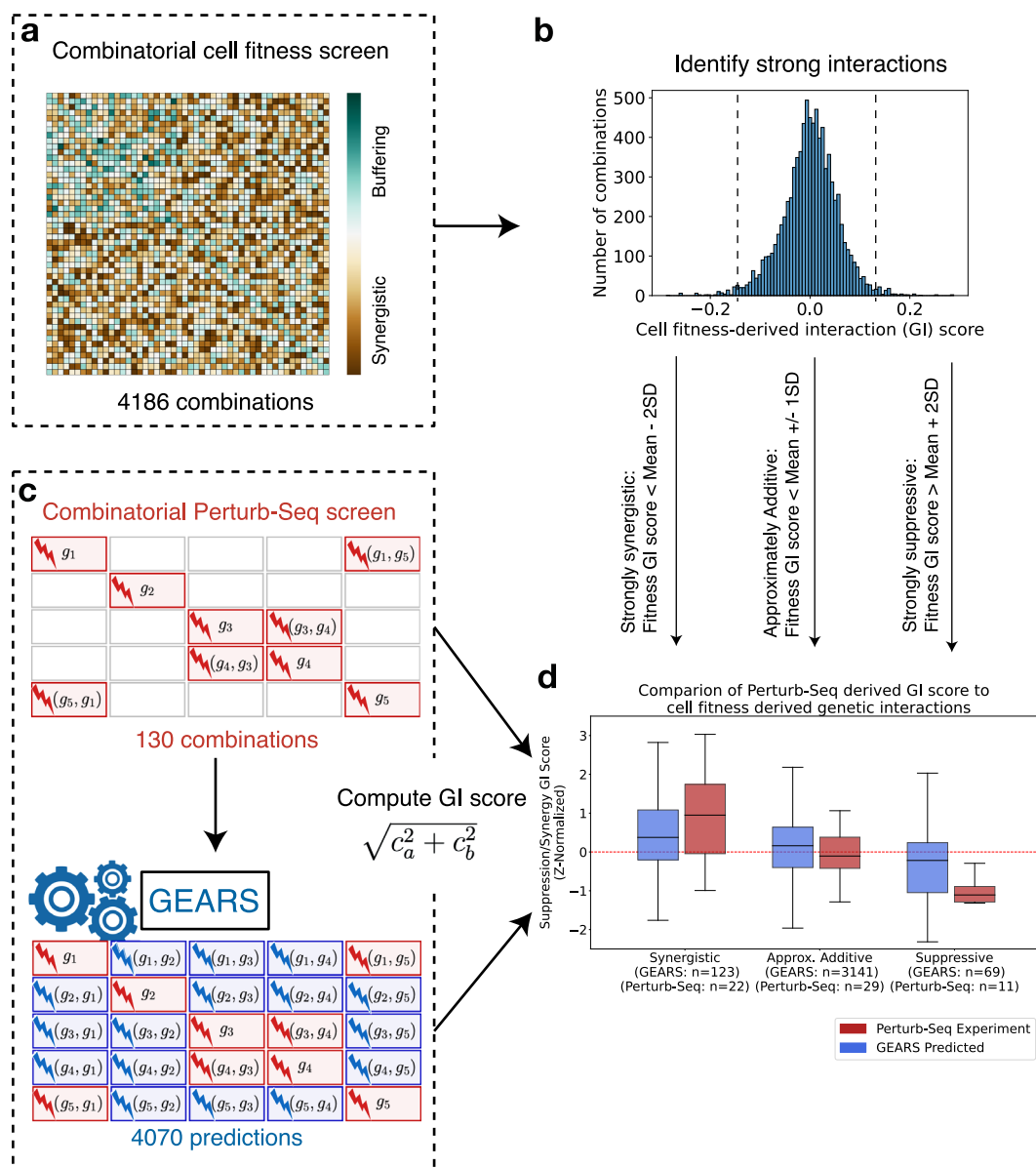
combination is exhibiting a specific GI subtype phenotype. Both GEARS and CPA were trained using a leave-one-out testing approach for each of the 131 combinations. The black dashed line represents the minimum and maximum of all 131 values and the black solid line represents the mean. **b,** Scatter plots of GI scores for all 131 two-gene combinatorial perturbations from that dataset. The x-axis shows GI scores computed using true postperturbation gene expression and the y-axis shows scores computed using predicted postperturbation gene expression. The top row shows predictions made by GEARS and the bottom row shows predictions made by CPA. R² refers to the coefficient of determination.



Extended Data Fig. 6 | See next page for caption.

Extended Data Fig. 6 | Model performance in predicting genetic interactions (GIs). **a**, Mean Square Error (MSE) in predicting non-additive combinatorial effects between the additive model which assumes that the effect of the combination is just the sum of the two known single-gene perturbation outcomes and GEARS predictions. MSE was measured on the 20 genes with the largest difference between true postperturbation expression following two-gene combinatorial perturbation and the additive prediction for that combination. GI subtypes (x-axis) were labelled without overlap as in Norman et al. 2019 (Synergy

n=30, Suppression n=12, Redundancy n=8, Neomorphism n=13, Epistasis n=9). Bar plots represent the mean and error bars correspond to 95% CI. **b**, Top 10 accuracy in predicting GIs: Model accuracy in predicting the set of 10 strongest interactions for each GI subtype as determined using true expression. Marker represents mean and error bar represents 1SD for the random model which performs 1000 draws (n=1000). For other models, predictions from 3 trained models were used (n=3). **c**, Precision and recall in predicting GIs (n=3).



Extended Data Fig. 7 | Validation of GEARs predicted genetic interaction (GI) map using combinatorial cell fitness screen. **a**, Combinatorial cell fitness screen data was used for all pairwise combination of 92 genes leading to 4186 unique combinations. Using cell fitness, interactions were quantified as synergistic or suppressive. **b**, Combinations showing the strongest cell fitness effects were used to validate GEARs predictions. **c**, Combinatorial Perturb-Seq data was available for 110 of these combinations. GEARs was trained on Perturb-Seq data to predict remaining 4076 perturbation outcomes. **d**, GEARs performs

similar to experimental Perturb-Seq data in predicting strong genetic interaction outcomes for both strongly synergistic and suppressive interactions identified using cell fitness measurements. GI scores are z-normalized within each modality for comparison. Centreline represents mean. Whiskers represent last data point within 1.5x interquartile range below the first quartile and above the third quartile, outliers not shown. The p-values were computed using a one-sided t-test comparing the means of the two distributions.

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see our [Editorial Policies](#) and the [Editorial Policy Checklist](#).

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a Confirmed

- | | | |
|-------------------------------------|-------------------------------------|--|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | The statistical test(s) used AND whether they are one- or two-sided
<i>Only common tests should be described solely by name; describe more complex techniques in the Methods section.</i> |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A description of all covariates tested |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals) |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
<i>Give P values as exact values whenever suitable.</i> |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated |

Our web collection on [statistics for biologists](#) contains articles on many of the points above.

Software and code

Policy information about [availability of computer code](#)

Data collection NA

Data analysis

All model code is available at <https://www.github.com/snap-stanford/GEARS>. Results can be reproduced using code available at https://github.com/yhr91/gears_misc.
Other packages used are: CellOracle (v 0.12.0) (<https://github.com/morris-lab/CellOracle>), CPA (<https://github.com/facebookresearch/CPA>, Accessed 09/2021), pySCENIC (v 0.12.0)

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

Only publicly available datasets were used:
Dixit et al., Cell (2016) GEO Accession Number GSE90063
Adamson et al., Cell (2016) GEO Accession Number GSE90546
Norman et al., Science (2019) GEO Accession Number GSE133344
Replegle et al. Cell (2022) Figshare DOI <https://doi.org/10.25452/figshare.plus.20029387.v1>

Jost et al. Nature Biotech. (2022) GEO Accession Number GSE132080
 Tian et al. Neuron (2019) GEO Accession Number GSE124703
 Replogle et al. Nature Biotech. (2020) GEO Accession Number: GSE146194
 Horlbeck et al. Cell (2018) GEO Accession Number: GSE116198

There are no figures with associated raw data.
 There are no restrictions on data availability

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see [nature.com/documents/nr-reporting-summary-flat.pdf](https://www.nature.com/documents/nr-reporting-summary-flat.pdf)

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	In case of experimental data, we used full published datasets, so sample size was defined by the studies that generated the data and any relevant post-processing. For computational experiments, we used 5 different random train-test data splits which is typical in the field (n=5). In the case of leave-one-out computational validation (e.g. for genetic interaction prediction), we ran the model 3 different times since the test set is fixed and variation between predictions is not significant (n=3)
Data exclusions	No data was excluded from the public datasets listed above
Replication	Computational experiments are all repeatable using the code provided.
Randomization	Not required as we were not running any new gene perturbation experiments.
Blinding	Not required as we were not running any new gene perturbation experiments.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> Antibodies
<input checked="" type="checkbox"/>	<input type="checkbox"/> Eukaryotic cell lines
<input checked="" type="checkbox"/>	<input type="checkbox"/> Palaeontology and archaeology
<input checked="" type="checkbox"/>	<input type="checkbox"/> Animals and other organisms
<input checked="" type="checkbox"/>	<input type="checkbox"/> Human research participants
<input checked="" type="checkbox"/>	<input type="checkbox"/> Clinical data
<input checked="" type="checkbox"/>	<input type="checkbox"/> Dual use research of concern

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> ChIP-seq
<input checked="" type="checkbox"/>	<input type="checkbox"/> Flow cytometry
<input checked="" type="checkbox"/>	<input type="checkbox"/> MRI-based neuroimaging