Single-Cell Response Modelling under Compositional Perturbations

Tianjun Yao

2022.04.12

Compositional perturbation autoencoder for single-cell response modeling

Mohammad Lotfollahi[1,3,⋆], Anna Klimovskaia Susmelj[2,5,⋆], Carlo De Donno [1,7,⋆⋆], Yuge Ji [1,⋆⋆], Ignacio L. Ibarra[1], F. Alexander Wolf [1,○], Nafissa Yakubova [2], Fabian J. Theis[1,3,4,6‡], David Lopez-Paz[2]

1 Helmholtz Center Munich – German Research Center for Environmental Health, Institute of Computational Biology, Neuherberg, Munich, Germany.

2 Facebook AI, 6 Rue Ménars, Paris, 75002, France

3 School of Life Sciences Weihenstephan, Technical University of Munich, Munich, Germany.

4 Department of Mathematics, Technical University of Munich, Munich, Munich, Germany.

5 Swiss Data Science Center, Zurich, Switzerland.

6 Wellcome Trust Sanger Institute, Wellcome Genome Campus, Hinxton, Cambridgeshire, UK.

7 Department of Stress Neurobiology and Neurogenetics, Max Planck Institute of Psychiatry, Munich, Bavaria, Germany.

⋆ These authors contributed equally to the work

⋆⋆ These authors contributed equally to the work

○ Present address: Cellarity, Inc., Cambridge, MA.

‡ Correspondence to fabian.theis@helmholtz-muenchen.de

Motivation

- Although recent developments facilitate single-cell RNA sequencing (scRNA-seq) in multi-sample experiments, these techniques require expensive library preparation and do not easily scale to large numbers of perturbations

- *experimental screening of all combinations is impossible*

- the development of computational tools to guide the exploration of the combinatorial space of perturbations is necessitated for OOD prediction.

Method

- Input:

i) Gene expressions per cell

ii) Perturbations: dose of drug to a particular cell

iii) covariates: cell types or Species etc.

- Output:

i) reconstructed gene expressions

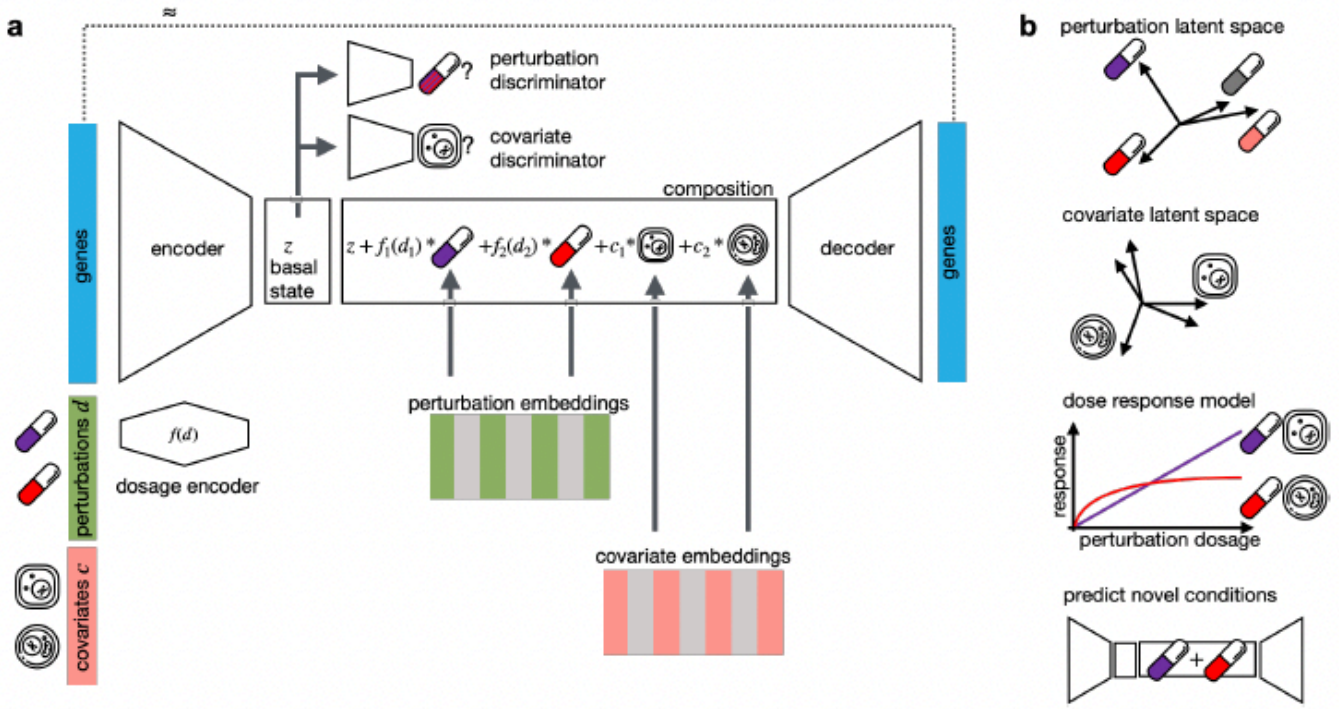


Figure 1 | Interpretable single-cell perturbation modeling using a compositional perturbation autoencoder (CPA). (a) Given a matrix of gene expressions per cell together with annotated potentially quantitative perturbations d and other covariates such as cell line, patient or species, CPA learns the combined perturbation response for a single-cell. It encodes the gene expression using a neural network into a lower dimensional latent space that is eventually decoded back to an approximate gene expression vector, aimed to be as close as possible to the original one. To make the latent space intepretable in terms of perturbation and covariates, the encoded gene expression vector is first mapped to a 'basal state', by feeding the signal to discriminators to remove any signal from perturbations and covariates. The basal state is then composed with perturbations and covariates - with potentially reweighted dosages - to reconstruct the gene expression. All encoder, decoder and discriminator weights as well as the perturbation and covariate dictionaries are learned during training. **(b)** Features of CPA are interpreted via plotting of the two learned dictionaries, interpolating covariate specific dose response curves and predicting novel unseen drug combinations.

Method

➢Assumption:

i) A basal state of gene expressions for each cell.

ii) Intepretability can be obtained through additive models.

➢Ideas:

i) Leveraging adversarial learning and GAN-like approach to learn the basal state for perturbed state, by utilizing a discriminator and generator(reconstructor).

ii) Alternatively train generator and discriminator.

Method

➢Training:

$\mathcal{D} = \{(x_i, d_i, c_i)\}_{i=1}^{N}$:dataset

- sample $(x_i, d_i, c_i) \sim \mathcal{D}$, minimize $\ell_i^d + \sum_j \ell_{i,j}^c$ by updating the parameters of $\hat{f}_d^{\text{adv}}$ and $\hat{f}_{c_{i,j}}^{\text{adv}}$, for all $j = 1, \ldots, K$;
- sample $(x_i, d_i, c_i) \sim \mathcal{D}$, minimize $\ell_i - \lambda \cdot (\ell_i^d + \sum_j \ell_{i,j}^c)$ by updating the parameters of the encoder $\hat{f}^{\text{enc}}$, the decoder $\hat{f}^{\text{dec}}$, the perturbation embeddings $\hat{V}^{\text{perturbation}}$, the covariate embeddings $\hat{V}^{\text{cov}_j}$ for all $j = 1, \ldots, K$, and the dose-response curve estimators $(\hat{f}_1, \ldots, \hat{f}_M)$.

$$z_i = z_i^{\text{basal}} + V^{\text{perturbation}} \cdot (f_1(d_{i,1}), \ldots, f_M(d_{i,M})) + \sum_{j=1,\ldots,K} V^{\text{cov}_j} \cdot c_{i,j}$$

: additive modelling of perturbed latent embedding z_i

Reconstruction loss: $\ell_i := \dfrac{\log s(\hat{f}_{\sigma^2}^{\text{dec}}(\hat{z}_i))}{2} + \dfrac{(\hat{f}_{\mu}^{\text{dec}}(\hat{z}_i) - x_i')^2}{2 \cdot s(\hat{f}_{\sigma^2}^{\text{dec}}(\hat{z}_i))},$

Discriminator loss:

$$\ell_i^d := \text{CrossEntropy}(\hat{f}_d^{\text{adv}}(\hat{z}_i^{\text{basal}}), d_i),$$
$$\ell_{i,j}^c := \text{CrossEntropy}(\hat{f}_{c_{i,j}}^{\text{adv}}(\hat{z}_i^{\text{basal}}), c_{i,j}), \quad \forall j = 1, \ldots, K.$$

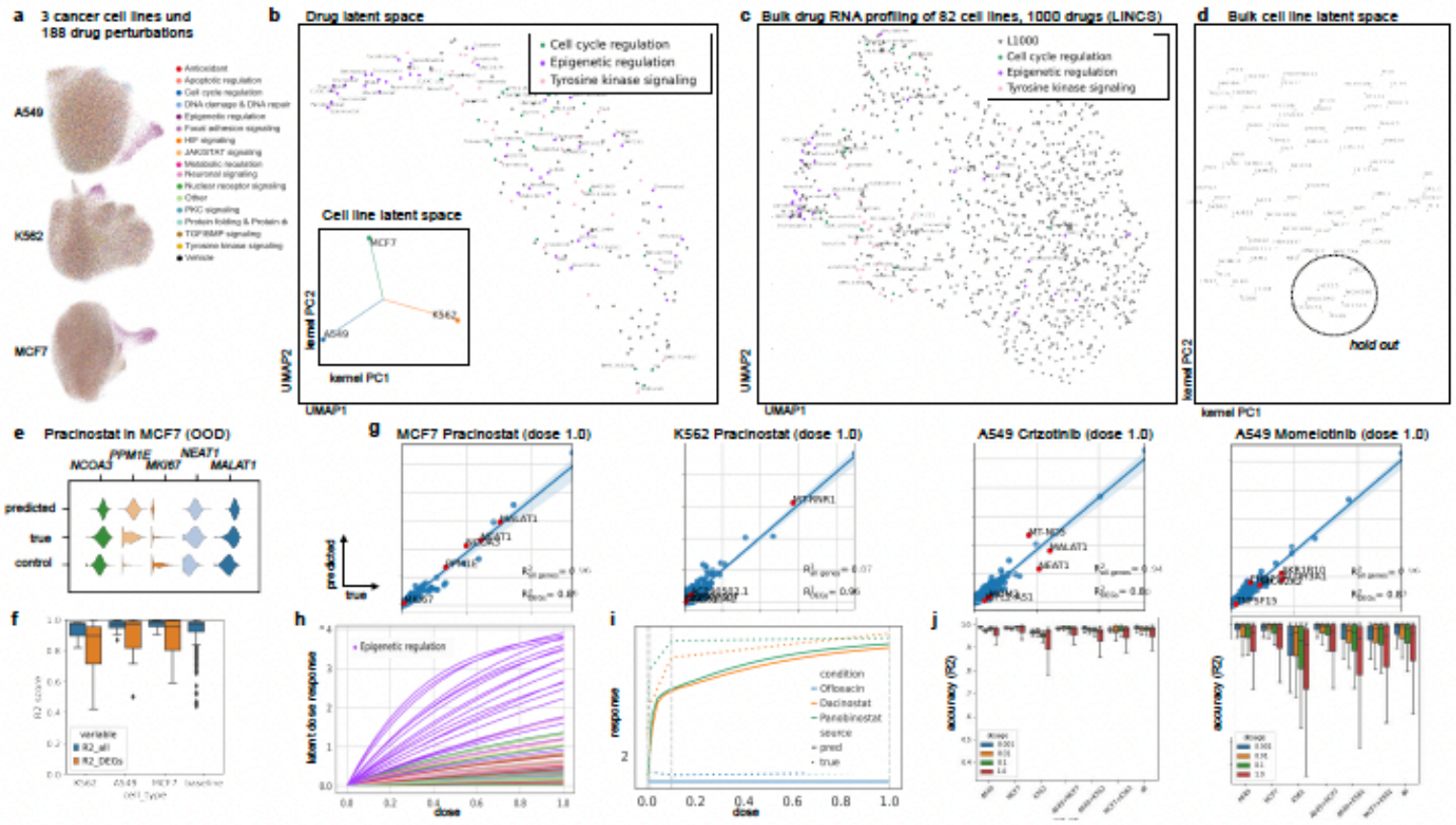
Testing:

- Compute the counterfactual perturbed state

$$\hat{z}_i' := \hat{z}_i^{\text{basal}} + \hat{V}^{\text{perturbation}} \cdot (\hat{f}_1(d_{i,1}'), \ldots \hat{f}_M(d_{i,M}')) + \sum_{j=1,\ldots,K} \hat{V}^{\text{cov}_j} \cdot c_{i,j}.$$

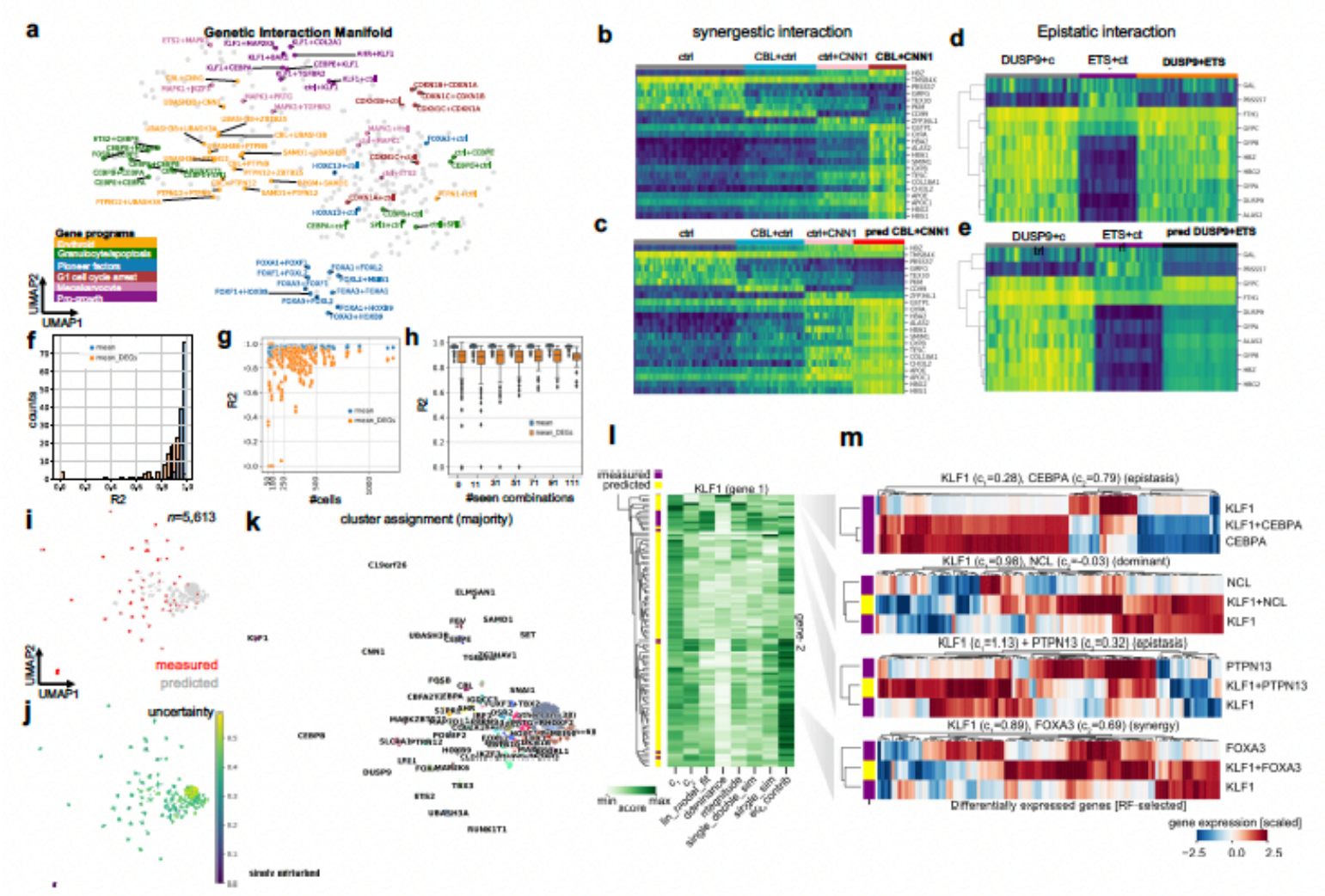
Result

- *epigenetic drugs, tyrosine kinase signaling and cell-cycle regulation compounds are clustered together by the model.*
- *Across difference datasets, the same epigenetic and tyrosine kinase signaling compounds are close to each other in the latent representation.*
- *cell lines from lung tissue are clustered together in response to perturbation.*
- *Able to extrapolate to the unseen OOD conditions with unexpected accuracy, CPA correctly infers the mean and distribution of these genes from Pracinostat*
- *CPA also does well in predicting genes with high mean expression in the OOD condition*



Result

- *latent genetic interaction manifold places GIs inducing known and similar gene programs close to each other*
- *synergistic interaction recapitulate similar pattern as real data inline*
- *genetic epistatic interaction between DUSP9 and ETS are also predicted accurately.*
- *The reported R2 values show robust prediction for most of the perturbations.*
- *overall prediction accuracy improves when the model is trained with more combinations, while it can fail to predict DEGs when trained with fewer combinations.*



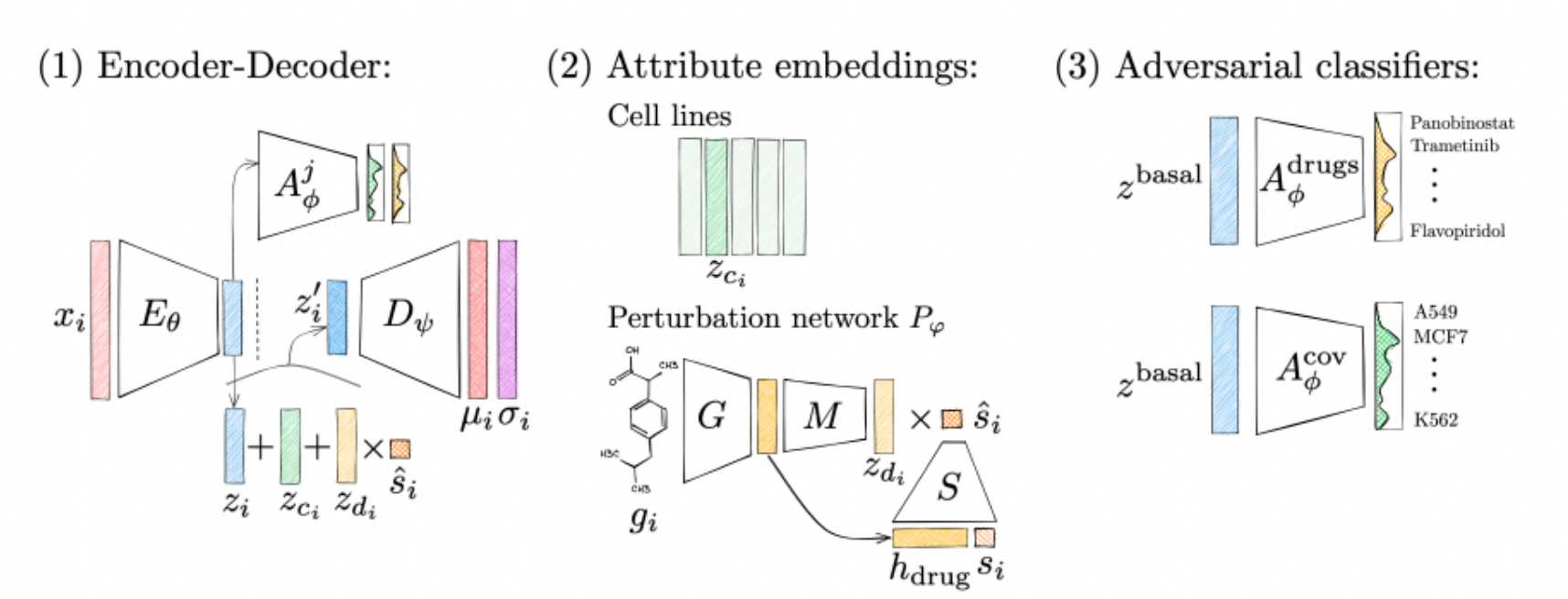
Predicting Cellular Responses to Novel Drug Perturbations at a Single-Cell Resolution

Leon Hetzel[*1,3], **Simon Böhm**[*3], **Niki Kilbertus**[2,4],
Stephan Günnemann[2], **Mohammad Lotfollahi**[1,5], **and Fabian Theis**[1,3]

{leon.hetzel, simon.boehm, niki.kilbertus}@helmholtz-muenchen.de
s.guennemann@tum.de, {mohammad.lotfollahi, fabian.theis}@helmholtz-muenchen.de

[1]Department of Mathematics, Technical University of Munich
[2]Department of Computer Science, Technical University of Munich
[3]Helmholtz Center for Computational Health, Munich
[4]Helmholtz AI, Munich
[5] Wellcome Sanger Institute, Cambridge

Method



➢Major difference from CPA:

i) Use a perturbation network to model perturbation conditions

ii) Perturbation response is modelled differently in the additive structure.

iii) Use a pretrained network for molecule encoder.

Method

➢Loss function(similar to CPA):

$$\mathcal{L}_{\text{rec}}(\theta,\psi) = N(x_i \,|\, \mu_i, \sigma_i) = \frac{1}{2}\left[\ln\left(D_\psi^{\sigma^2}(z_i')\right) + \frac{\left(D_\psi^{\mu}(z_i') - x_i\right)^2}{D_\psi^{\sigma^2}(z_i')}\right] \text{ with } z' = E_\theta(x) + z_{\text{attribute}} \,,$$

$$\mathcal{L}_{\text{class}}^{\text{drugs}} = \text{CE}\big(A_\phi^{\text{drug}}(z_i)\big), d_i) \quad \text{and} \quad \mathcal{L}_{\text{class}}^{\text{cov}} = \text{CE}\big(A_\phi^{\text{cov}}(z_i), c_i\big)$$

$$\mathcal{L}_{\text{AE}}(\theta,\psi,\varphi|\phi) = \mathcal{L}_{\text{rec}}(\theta,\psi,\varphi) - \lambda_{\text{dis}} \sum_j \mathcal{L}_{\text{class}}^j(\theta \,|\, \phi) \quad \text{and}$$

$$\mathcal{L}_{\text{Adv}}(\phi \,|\, \theta) = \sum_j \mathcal{L}_{\text{class}}^j(\phi \,|\, \theta) + \lambda_{\text{pen}} \mathcal{L}_{\text{pen}}^j(\phi) \,,$$

➢Perturbation network:

$$\hat{s}_i \times z_{d_i} = P_\varphi(g_i, s_i) = S(h_{d_i}, s_i) \times M(h_{d_i}) \text{ with } h_{d_i} = G(g_i)$$

g_i: molecule representation; G: molecule encoder; M: permutation encoder; S: dosage encoder

s: dosage value

Results

➢Evaluation metric: r^2 to measure the correlation for regression task.

Table 1: Comparison of multiple models on their performance on generalisation to unseen drug-covariate combinations for dosage values of 1 μM and 10 μM.

Dose	Model	$\mathbb{E}[r^2]$ all	$\mathbb{E}[r^2]$ DEGs	Median r^2 all	Median r^2 DEGs
1 μM	Baseline	0.69	0.51	0.82	0.62
	scGen	0.73	0.59	0.77	0.68
	CPA	0.72	0.54	**0.86**	0.67
	chemCPA	0.74	0.60	**0.86**	0.66
	chemCPA pretrained	**0.77**	**0.68**	0.85	**0.76**
10 μM	Baseline	0.50	0.29	0.48	0.12
	scGen	0.62	0.47	0.66	0.49
	CPA	0.54	0.34	0.52	0.26
	chemCPA	0.71	0.58	0.77	0.64
	chemCPA pretrained	**0.76**	**0.68**	**0.82**	**0.79**

Baseline: without adding any perturbation information into the modelling process

Evaluation: average across all drugs and cell lines

Results

➢Evaluation metric: r^2 to measure the correlation for regression task.

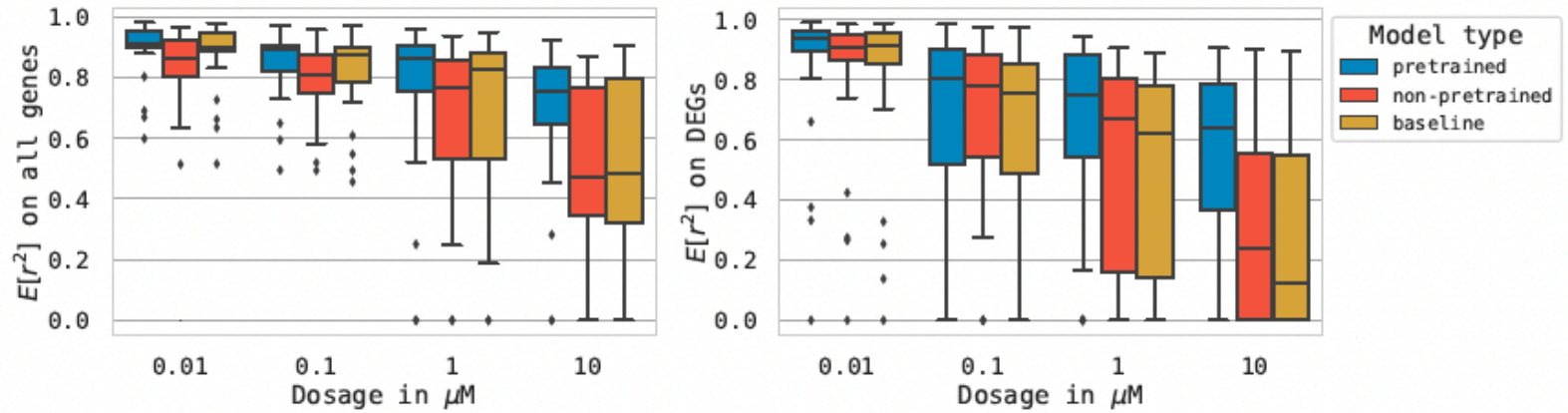


Figure 2: Performance of chemCPA on both the complete gene set (977 genes) and the compound specific DEGs (50 genes). In both cases, the pretrained model shows the best performance. At 10 μM on the DEGs, more than 50% of the predictions have an r^2 score > 0.6 while the baseline's median is below 0.2.

Baseline: without adding any perturbation information into the modelling process

Experiment settings: r^2 average across all drugs and cell lines for unseen compounds on shared gene set.

Pretrained models benefits the prediction significantly.

Results

➢Evaluation metric: r^2 to measure the correlation for regression task.

Table 3: We show the performance of chemCPA on the extended gene set. Since drug effects are stronger for high dosages, we present scores for a dosage value of 10 μM.

Model	$\mathbb{E}[r^2]$ all	$\mathbb{E}[r^2]$ DEGs	Median r^2 all	Median r^2 DEGs
Baseline	0.37	0.19	0.16	0.00
chemCPA	0.46	0.22	0.35	0.00
chemCPA pretrained	**0.69**	**0.47**	**0.79**	**0.62**

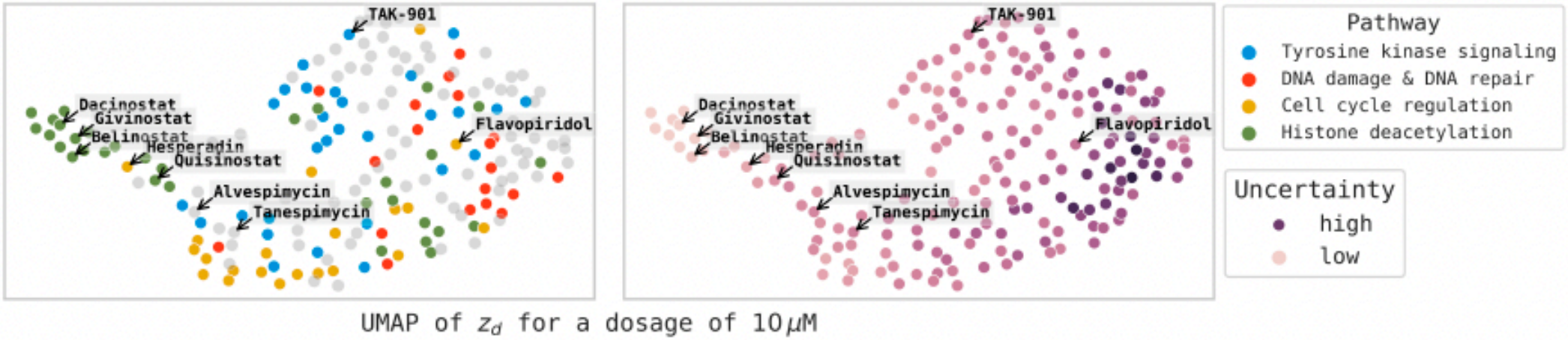
Baseline: without adding any perturbation information into the modelling process

Experiment settings: r^2 average across all drugs and cell lines for unseen compounds on shared gene set.

include HVGs to account for the technological difference between bulk and single-cell and to capture the variance of single-cell data. the 977 genes present in both datasets are extended with 1023 HVGs of the sci-Plex3 data.

Results

➢Evaluation metric: r^2 to measure the correlation for regression task.



the perturbation embeddings are clustered according to some of the pathways.

MultiCPA: Multimodal Compositional Perturbation Autoencoder

Kemal Inecik [1 2] **Andreas Uhlmann** [3] **Mohammad Lotfollahi** [1 *] **Fabian Theis** [1 3 *]

Method

- Use both gene expression and surface protein for modelling process
- Two architectures: i) concatenation model; ii) POE (Product-of-Expert) model.
- Still assuming a basal state, and use adversarial training for disentanglement.
- In addition to gene expression value x_G, also consider reconstruction of protein state.

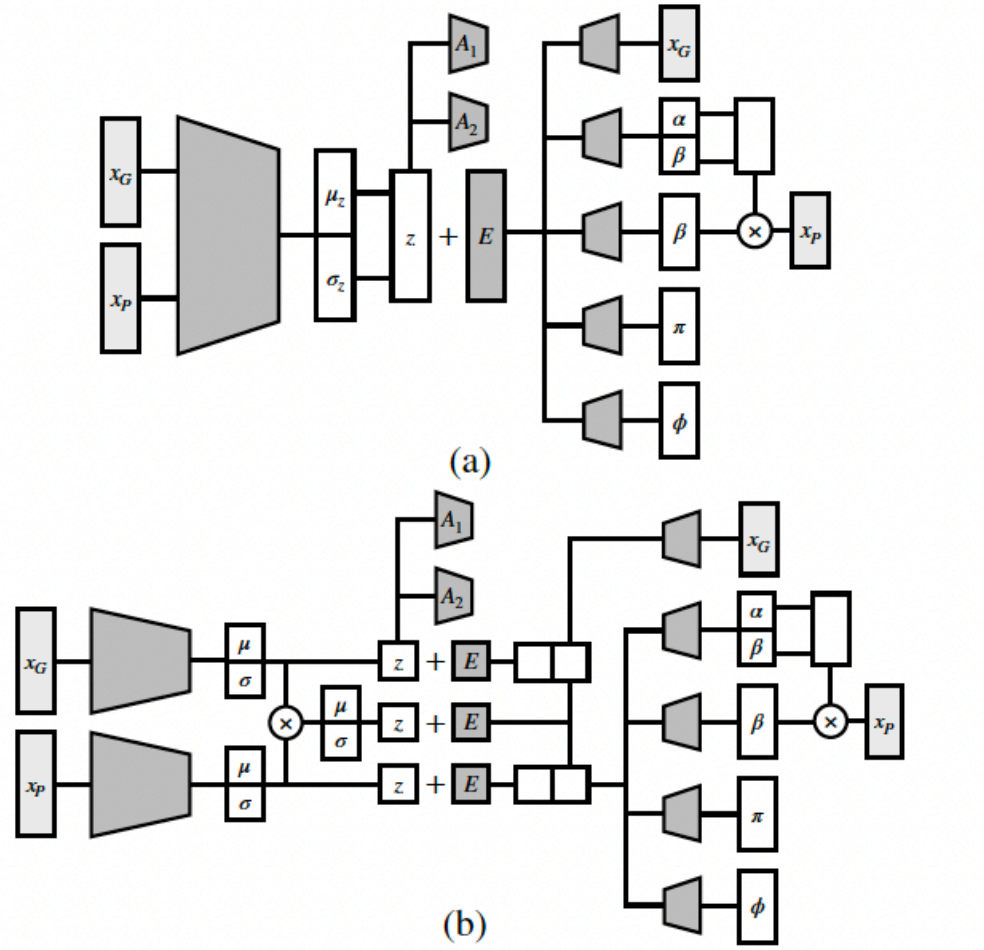


Figure 1. Overview of proposed MultiCPA architectures, where A_i denotes adversarial discriminator networks, E denotes separate perturbation and covariate embeddings. Multimodal integration by a) concatenation mixture module. b) PoE mixture module.

Method

➢Loss function for gene:

$$\mathcal{L}_{\mathrm{G}} = \mathrm{NB}(x;\mu,\theta) = \frac{\Gamma(x+\theta)}{\Gamma(x+1)\Gamma(\theta)}\left(\frac{\theta}{\theta+\mu}\right)^{\theta}\left(\frac{\mu}{\theta+\mu}\right)^{x}$$

$$\mathcal{L}_{\mathrm{M},1} = \mathrm{KL}(\mathcal{N}(\mu_{joint},\sigma_{joint})||\mathcal{N}(0,1)).$$

➢Loss function for protein:

$$\mathcal{L}_{\mathrm{P}} = \pi\mathrm{NB}(x;\mu_b,\theta) + (1-\pi)\mathrm{NB}(x;\mu_f,\theta)$$
$$\mathcal{L}_{\mathrm{M},2} = \mathrm{KL}(\mathcal{N}(\alpha,\beta)||\mathcal{N}(\alpha_{prior},\beta_{prior}))$$

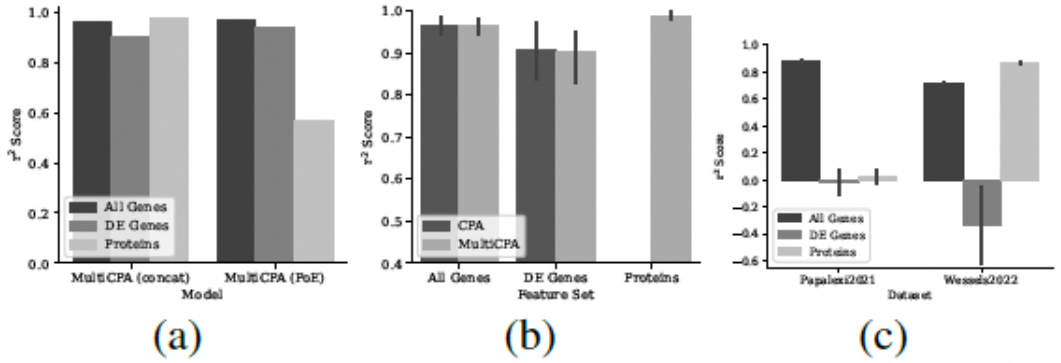
u_b:background mean u_f:foreground mean

➢Total loss:

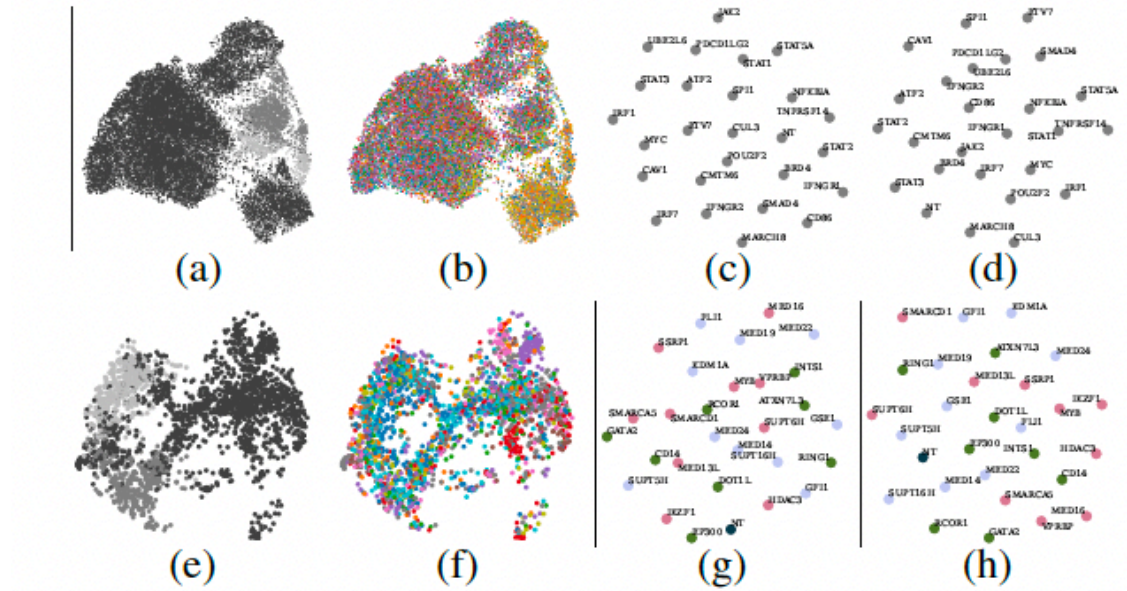
$$\mathcal{L}_{\mathrm{MultiCPA}} = \mathcal{L}_{\mathrm{G}} + \mathcal{L}_{\mathrm{P}}w_1 + (\mathcal{L}_{\mathrm{M},1} + \mathcal{L}_{\mathrm{M},2})w_2 - (\mathcal{L}_{\mathrm{A},1} + \mathcal{L}_{\mathrm{A},2})w_3$$

Result

- Upper figure:
  - a) Concatenation model outperform POE model in predicting protein data(0.98 vs. 0.57)
  - MultiCPA performs similarly to CPA in terms of counterfactual reasoning.



- Lower figure:
  - not able to group the perturbations in both datasets in a consistent fashion, suggesting a dataset specific variation or potential issues that still need to be accounted
  - both models could successfully extract perturbation information from the input datasets



Thank You