RESEARCH ARTICLE

DeepCellState: An autoencoder-based framework for predicting cell type specific transcriptional states induced by drug treatment

Ramzan Umarov[1]*, **Yu Li**[2], **Erik Arner**[1,3]*

1 Graduate School of Integrated Sciences for Life, Hiroshima University, Higashi-Hiroshima, Japan, **2** Department of Computer Science and Engineering (CSE), The Chinese University of Hong Kong (CUHK), Hong Kong, People's Republic of China, **3** Laboratory for Applied Regulatory Genomics Network Analysis, RIKEN Center for Integrative Medical Sciences, Yokohama, Kanagawa, Japan

* umarov@hiroshima-u.ac.jp (RU); erik.arner@riken.jp (EA)

OPEN ACCESS

Citation: Umarov R, Li Y, Arner E (2021) DeepCellState: An autoencoder-based framework for predicting cell type specific transcriptional states induced by drug treatment. PLoS Comput Biol 17(10): e1009465. https://doi.org/10.1371/journal.pcbi.1009465

Editor: Magnus Rattray, University of Manchester, UNITED KINGDOM

Received: May 3, 2021

Accepted: September 20, 2021

Published: October 5, 2021

Data Availability Statement: Trained models, code to generate them, and code for analysis and figures described in this study are available at the following GitHub repository: https://github.com/umarov90/DeepCellState. All data used in training and validation is available through public databases referenced in this study.

Funding: The author(s) received no specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

Abstract

Drug treatment induces cell type specific transcriptional programs, and as the number of combinations of drugs and cell types grows, the cost for exhaustive screens measuring the transcriptional drug response becomes intractable. We developed DeepCellState, a deep learning autoencoder-based framework, for predicting the induced transcriptional state in a cell type after drug treatment, based on the drug response in another cell type. Training the method on a large collection of transcriptional drug perturbation profiles, prediction accuracy improves significantly over baseline and alternative deep learning approaches when applying the method to two cell types, with improved accuracy when generalizing the framework to additional cell types. Treatments with drugs or whole drug families not seen during training are predicted with similar accuracy, and the same framework can be used for predicting the results from other interventions, such as gene knock-downs. Finally, analysis of the trained model shows that the internal representation is able to learn regulatory relationships between genes in a fully data-driven manner.

Author summary

A large number of gene expression profiles across different cell types are available, however many drug-cell combinations have not been profiled. Motivated by the need for accurate methods for prediction of cell type specific drug responses, we developed DeepCellState, a deep learning framework, with the goal of predicting the response in a given cell type based on the response in another cell type. Training the method on the largest available database for transcriptional response to drug perturbations, LINCS, we observed that the method can predict with high accuracy the cell type specific response for treatment with drugs not seen by the method, with improved accuracy as we generalized the method from two to multiple cell types. Encouragingly, the method performed well even when the response from completely unseen cell types were used as input. We further confirmed the robustness of our results through validation with data independently generated on other expression profiling platforms. Analysis of the learned models revealed although the training is completely data driven and uses no prior knowledge about regulatory relationships between genes, the network itself is biologically interpretable and captures interactions between transcription factors and the targets they regulate.

Introduction

The transcriptional response to drug treatment is cell type specific, with some drugs eliciting similar effects across lineages and others evoking a range of responses depending on the cell type [1,2]. High throughput profiling of the transcriptional effects of drug treatment has proven to be useful for analysis of drug mode of action [3], drug repurposing [4], and predicting off-target effects from drug treatment [5]. Although large repositories of gene expression profiles for a multitude of drug treatments in multiple cell types are available [6,7], it is not combinatorially feasible to profile all the existing drugs in all the relevant cell types, motivating a need for methods that can accurately predict cell type specific drug responses.

An autoencoder neural network is an unsupervised learning algorithm that finds an efficient compact representation of data by compressing and then reconstructing the original input. The primary goal of an autoencoder is dimensionality reduction. Autoencoders can have different architectures. However, the crucial feature is a bottleneck layer (latent layer), which has lower dimensionality compared to the input layer. Because of the bottleneck, only important features are captured by the model. Combining this property together with the addition of noise to the input allows for the construction of denoising autoencoders to build robust models from high-dimensional data. A number of recent studies have successfully applied autoencoders to biological problems, where deep autoencoders were used to denoise single cell RNA-seq data sets [8,9], analyze [10,11] and predict [12,13] gene expression, and to study the transcriptomic machinery [14]. Autoencoders have been applied to perturbation response modeling as well, focusing on single cell data [15], where for each perturbation, a large number of expression profiles are available with relatively low variance within sets of profiles from the same perturbation.

A particular application of autoencoders is DeepFake technology, mainly used in image and video processing applications to generate synthetic media where the likeness of one person is replaced by that of another one by training an autoencoder to compress the original input into a lower dimension latent space [16]. The same encoder part is used to compress media depicting two or more people, whereas separate decoders are used to decompress the likeness of each person (S1 Fig). This enables similar facial expressions or actions to be encoded in a similar way in the latent space while at the same time allowing for the reconstruction of person-specific facial details on the decoder side.

Hodos et al. [2] used matrix completion approach to predict cell-specific drug perturbation profiles using available expression data. Their method arranges existing profiles into a three-dimensional array (or tensor) indexed by drugs, genes, and cell types, and then uses either local or global information to predict unmeasured profiles. The developed tensor-based method was reported to achieve an average correlation of 0.54 using multiple cell types as input. One can also apply the new framework of deep dictionary learning [17] or deep Boltzmann machine [18] to generate the missing cell type/perturbation combinations. Considering that there is still large room for improvement for transcriptome perturbation analysis and given the previous success of autoencoder-based approaches for modeling gene expression data, we developed DeepCellState, an autoencoder-based framework inspired by DeepFake architecture, for predicting cell type specific transcriptional drug response. Using a common encoder and separate decoders for each cell type (Fig 1A), the method accurately predicts the response in one cell type based on the response in another, with prediction accuracy increasing as the framework is generalized to multiple cell types. DeepCellState achieves significant improvement over the baseline and alternative deep learning approaches. Cell types not used in training can also be used as the basis for prediction, and the system is also able to predict the response of the entire drug families not seen by the network. Additionally, the same framework can be applied to predict the effects of loss of function (LoF) experiments. Analyzing the trained network, we find that it captures physical interactions between transcription factors and the target genes they regulate.

Results

DeepCellState network architecture for predicting cell type specific drug response

The network architecture of DeepCellState, featuring a common encoder and cell type specific decoders, is shown in Fig 1B. The encoder part uses two dense layers, with dropout applied to the input for increased generalization capabilities by forcing the model to denoise the input [19]. This is followed by a latent vector, which in turn is the input for the decoder, consisting of two layers with the same dimensions as the dense layers in the encoder, but in reverse order. The encoder side is trained using transcriptional drug response data from multiple cell types, whereas distinct decoders are trained for each cell type separately. Additional details about the network architecture and training strategy are provided in **Methods**.

Initial training and evaluation of model on two cell types

We initially developed the framework for two cell types, retrieving 12737 and 12031 transcriptional profiles from the LINCS database [6] measuring the drug perturbation response in MCF-7 and PC-3 cell lines respectively, to use for initial training and evaluation (**Methods**). The transcriptional response profiles, measured at 978 “landmark genes” on the L1000 platform, were averaged across treatment time and doses in order to decrease noise [3]. We only kept profiles of drugs that were available for both MCF-7 and PC-3 for the convenience of implementation, resulting in 1750 averaged profiles for each cell type.

A naïve baseline method for predicting the transcriptional response in a cell type after treatment with a specific drug is to predict that the response to that drug will be the same as in another cell type. After training our network and using 10-fold cross-validation, holding out 10% of drugs, we obtained an average Pearson correlation coefficient (PCC) of 0.60 between predicted and actual treatment response to drugs not seen by the network, a significant improvement ($p<1e-300$, t-test) compared to the baseline average correlation of 0.28 between the responses in the input and output cell types (Figs 2A, 2B, 2C, 2D and S2), corresponding to an average fold-increase of 2.14 over baseline.

We wanted to explore how the results will change if the number of drugs was different for PC-3 and MCF-7 cell types. We repeated the 10-fold cross-validation 3 times: with 70% of PC-3 training profiles removed, with 70% of MCF-7 training profiles removed, and 70% of both PC-3 and MCF-7 training profiles removed. In each case, we measured MCF-7 profiles prediction accuracy using PC-3 profiles as input. As expected, in the latter case, the performance was the worst, dropping to 0.50 PCC. Interestingly, for the other two experiments similar performance was obtained. For the case when MCF-7 training data was reduced, 0.55 PCC was

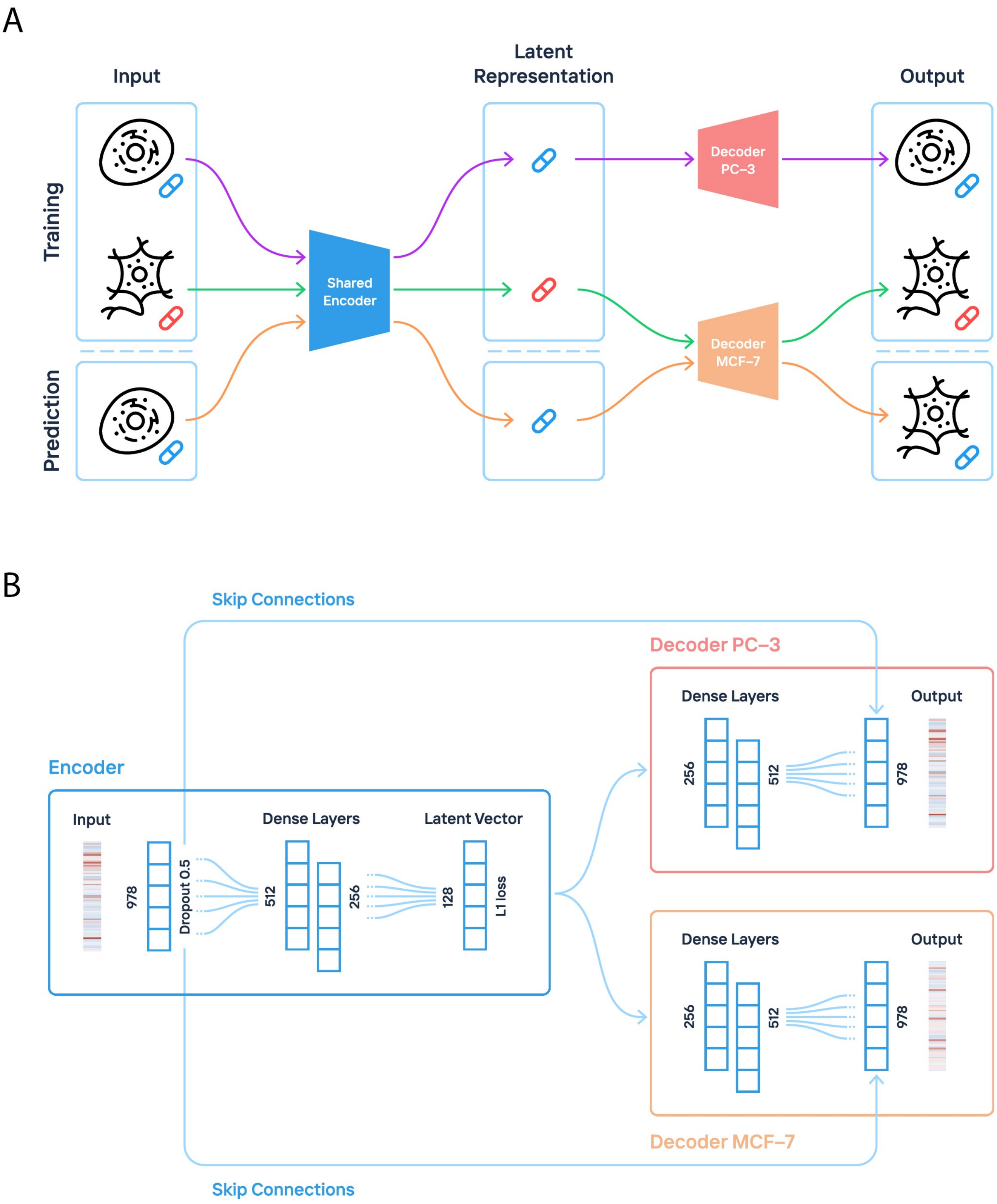

Fig 1. Overview of the proposed framework. (**A**): Training and prediction procedures of DeepCellState. Transcriptional profiles are encoded by a shared encoder that captures the drug response in a cell neutral way. The latent representations of the drug responses are decoded by cell-specific decoders, which reconstruct the original input in a cell type specific way. After the shared encoder and decoders are trained, the response to a drug in a cell type can be

predicted by using the drug's response in another cell type. (**B**): The architecture of the encoder and the decoders used in DeepCellState. Dropout is applied to the input layer, forcing the model to denoise the input. The input is encoded as a vector of size 128 with a sparsity constraint enforced by L1 regularization. The model also employs skip connections from the dropout layer directly to the output layer, allowing some of the input to be directly copied to the output during training.

https://doi.org/10.1371/journal.pcbi.1009465.g001

achieved, while for the case when 70% of PC-3 profiles are dropped, the performance decreased to 0.56. These results suggest that the output profile's cell type data is slightly more important than the number of training profiles which are used for the model input.

To make sure that the training was not biased by being exposed to drugs with similar targets, we next revised the testing strategy to hold out entire drug families with the same modes of action [20], yielding similar results (S3A Fig and S1 Table and **Methods**).

Generalization of the method to multiple cell types

Proceeding to generalize the method to multiple cell types, we incrementally added transcription profiles from A375, HA1E, HT29, HELA, and YAPC cells (1796, 1796, 1750, 1570, and 1570 number of profiles respectively after averaging across treatment times and dosages), obtaining increased performance when the prediction was averaged over multiple input cell types compared to the two cell types case (PCC = 0.66, p = 1.88e-59, S3B Fig and S2 Table). The total time to train using this much larger data was only one hour and forty minutes, increasing less than two times compared to the two cell type model. We also tested the prediction capability using completely unseen cell types as input. We have used A375 profiles as input to predict MCF-7 profiles by a model which was trained using only MCF-7 and PC-3 profiles. This resulted in a lower performance compared to using PC-3 profiles as input, yet the results were still significantly better than baseline (S3C Fig, PCC = 0.49 compared to baseline PCC = 0.28, $p<1e\text{-}300$).

Evaluation of trained model on external data and application of transfer learning

To further validate the performance of DeepCellState, we evaluated its predictive capabilities on independently generated data sets on expression profiling platforms other than the L1000 platform used for the training data. In the first dataset we used, the expression profiles of four statins were measured in HepG2, MCF-7, and THP-1 cells [21] using CAGE technology. Using HepG2 profiles as input and predicting MCF-7 response, the DeepCellState predictions showed a correlation of 0.43 with the true response in MCF-7 cells (1.64-fold over baseline, p = 0.037, S3 Table and Fig 2E). Next, we tested our method using data from a study measuring the response to 15 anticancer agents [22] on Affymetrix Human Genome U133A 2.0 Array platform, achieving 0.41 PCC (Fig 2F) compared to 0.28 PCC for baseline (1.46-fold improvement, p = 0.026, S4 Table). The common drugs from the two external data sets were removed from the training and validation sets. More details on the data processing are in **Methods**.

Performance on the anticancer external dataset could be further improved through transfer learning, performing additional training of our model on a subset of the external data set. To this end, we used the model trained on the LINCS data and additionally trained it on 14 drugs from the anticancer set to predict the effects of treatment with the 15th drug. This was repeated 15 times, each time leaving out one drug, resulting in an average performance of 0.56 PCC (p = 4.04e-06 compared to baseline). This result suggests that the transfer learning approach is useful to complete a small drug response data set which is not large enough to train the deep learning model by itself.

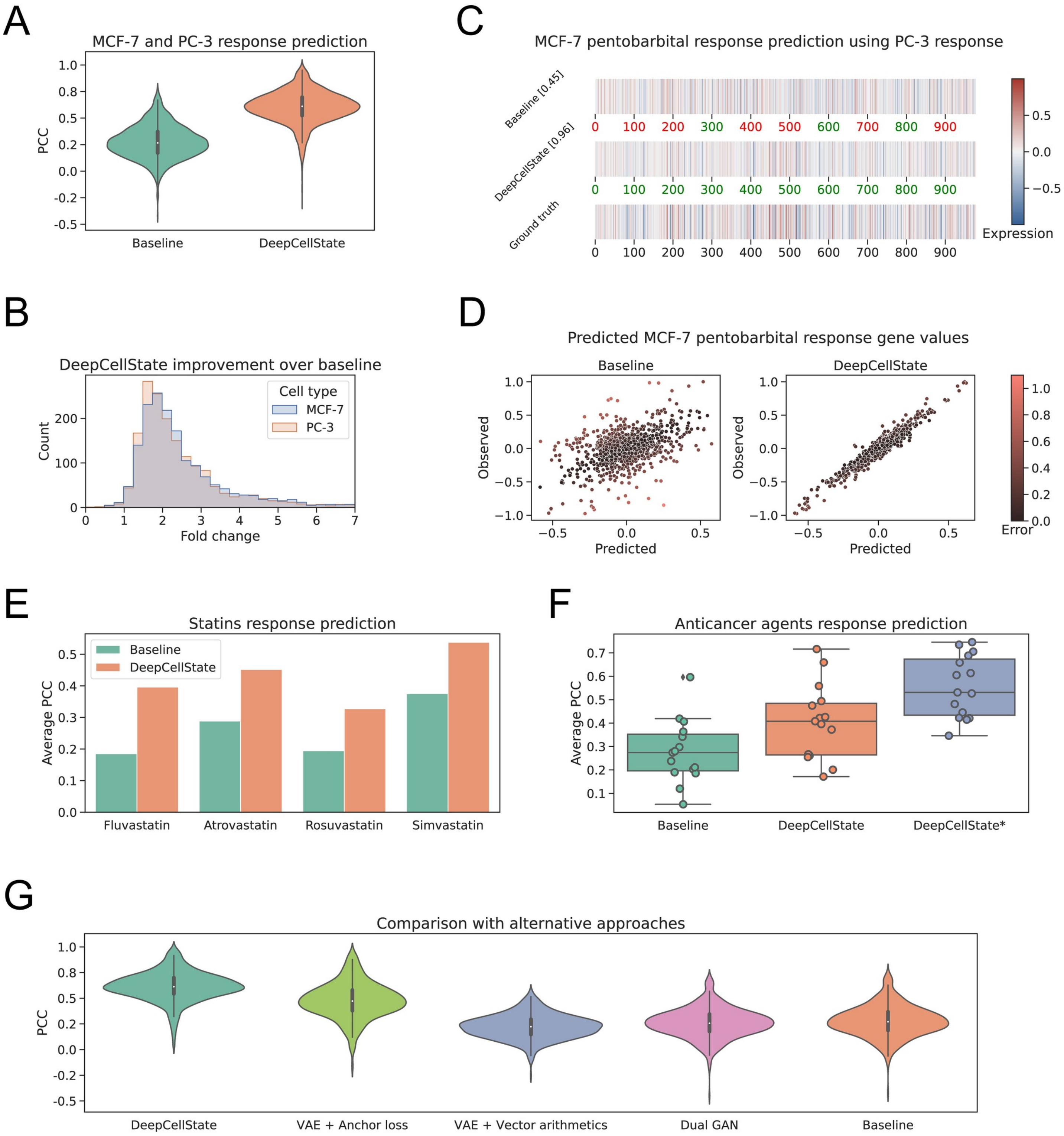

Fig 2. Evaluation of the DeepCellState performance. (**A**): 10-fold cross-validation results. DeepCellState performance compared to the baseline. (**B**): Distribution of fold-change in DeepCellState performance over baseline for predicting MCF-7 (blue) and PC-3 (red) response. (**C**): From top to bottom pentobarbital response in PC-3 (baseline), MCF-7 pentobarbital response predicted by DeepCellState based on the PC-3 response, and the observed MCF-7 response (ground truth). (**D**): Scatter plot of individual gene values predicted for pentobarbital response compared to the observed response in MCF-7 cells. (**E**): Comparison of DeepCellState and baseline performance for prediction of statin response from an external data set. (**F**): Comparison of predicted anticancer drug responses. HepG2 profiles are used as input for the model while the output profiles are for MCF-7 cells. DeepCellState* represents results obtained by applying transfer learning: performing additional training on 14 drugs

and predicting one which is repeated 15 times each time leaving out one drug. (**G**): Comparison with alternative methods for paired data modeling, using PC-3 responses to predict MCF-7 samples.

https://doi.org/10.1371/journal.pcbi.1009465.g002

Prediction of transcriptional effects of loss of function

We further hypothesized that the DeepCellState approach may be useful for predicting the response to cellular perturbations other than drug treatment. To test this, we added transcriptional profiles from loss of function (LoF) experiments from the same database to the training set. Using the updated set, we obtained an average correlation of 0.56 (corresponding to an average fold-increase over baseline of 1.6, $p<1e-300$, S3D Fig) on the LoF experiments, the lower performance compared to drug perturbations likely due to that the number of available profiles for LoF was substantially lower than the number of drug treatment profiles (10230 LoF profiles in total compared to 24768 drug perturbation profiles for MCF-7 and PC-3 before averaging).

Comparison with alternative approaches

We benchmarked DeepCellState against other potential methods to predict the cell response given the response in another cell. For this purpose, we focused on paired MCF-7 and PC-3 samples, with 30% data used for the test set. Recently, variational auto-encoders (VAEs) were used to deal with the paired biological data [23]. Anchor loss was used in this approach to ensure that paired samples are encoded closely in the latent space. Generative adversarial networks (GANs) were also applied to paired and semi-paired biological data [24,25]. The authors of MAGAN [25] proposed a dual GAN setup with a special correspondence loss for the labeled data. In the dual GAN framework, each generator learns a mapping from one type of data to another in order to generate increasingly realistic output that cannot be distinguished from real data by a discriminator.

scGen [15] was proposed to predict single cell perturbation responses based on VAE architecture, where an encoder is used to learn the semantic embedding of both the normal cell state and the stimulated state. The difference between such embeddings could then be considered as the embedding of the drug stimulation. When applying the method onto new cell types, the authors proposed to perform latent space vector arithmetic, adding the stimulation embedding to the normal cell embedding, and to pass the resulting vector through the decoder, obtaining the predicted responses. Since we have multiple drugs and the amount of data for each drug is limited, we modified the method to learn the difference between different cell types (**Methods**).

Comparison of the tested methods on the benchmark dataset is shown in Fig 2G and S5 Table. We were unable to obtain a good performance using the dual GAN approach (PCC = 0.26), likely due to the training data being too small. The data for each cell type has very high variance, making it hard for a discriminator to learn the difference between simulated and real cell response. scGen achieved an average correlation of 0.22, the main reason being that scGen is designed specifically for one kind of stimulation. However, in the problem we are trying to solve, multiple drugs and multiple cells need to be handled simultaneously and scGen showed limited capacity in handling this multi-task problem. The VAE using anchor loss obtained significant improvement over the baseline (PCC = 0.48 compared to baseline PCC = 0.28), however, the profiles predicted by DeepCellState were considerably more accurate (PCC = 0.61, $p<2.09e-40$). After these experiments, we attempted to incorporate VAE as well as GAN into the DeepCellState approach, but the improvement was not significant enough to justify the added complexity.

We next performed comparison with methods from study by Hodos et al (DNPP and FaLRTC) [2] using all the cell types and performing 10-fold cross-validation. Since VAE and GAN methods are designed specifically for translating between two domains, we could not include them in this comparison. We have achieved 0.53 PCC for DNPP, 0.49 for FaLRTC, and 0.65 for our approach (S4 Fig). We repeated the comparison using the processed data from the original study which contains considerably more cell types (71 vs 30). Our method still outperformed DNPP and FaLRTC, achieving 0.56 PCC, while DNPP and FaLRTC both achieved 0.46 (S5 Fig). The details of this comparison are provided on our GitHub repository.

The tested deep learning approaches had similar running time to our method. Since FaLRTC does not require any training, it can be run on the CPU and produces results for all the data in less than fifteen minutes. Our method took about three hours for the same data, however after the training is complete, the inference time for a new drug profile is only a fraction of a second

Model performance analysis

We next investigated more deeply the relationship between the number of input profiles used in training and the prediction performance. Results showed that significant improvement over baseline was possible already for a small number of training examples, with higher performance as the number of training examples increased (Fig 3A). To study the impact of the latent layer size on the performance of DeepCellState, we tested five different layer sizes (8, 16, 32, 64 and 128 neurons) and trained our model 10 times for each size. Average results for the test set (shown in Fig 3B) showed that performance started to drop significantly at a latent layer size below 32.

To pinpoint the benefit of the skip connection and regularization, we trained a simple model without a dropout layer, L1 regularization, and skip connection. For the two cell type case, the drop in performance was modest (PCC = 0.58, p<3.29e-9). However, for the external datasets the drop in performance was much more significant. For the statin data, we got an average PCC decrease of 0.04, and for the anticancer data the decrease was even higher, about 0.09 on average across three runs. We believe there are two reasons for this. Firstly, skip connection helps the model to deal with situations when it cannot predict a gene value. In this case, the value from the input layer can be copied directly. Secondly, without the regularization the model captures the technical bias of the LINCS data, performing significantly worse when data generated by a different platform is used. Overall, these results confirm that skip connection and regularization are beneficial for our model. Finally, we tested the benefits of adding the regularization techniques one by one and the results have reinforced our previous conclusion (S6 Table).

Biological interpretation of trained model

We set out to analyze general and cell type specific regulatory properties captured by the trained model, expecting that the latent vector itself should capture a general response to the input perturbations, whereas the decoder parts should represent behavior related to functionality specific to each cell type. To this end, we computed the PCC between the output of the nodes in the latent layer for the drugs in the test set between MCF-7 and PC-3 cells. Indeed, although considerably different, the inputs from the same drug in the two cell types consistently yielded similar output in the latent layer as illustrated in a t-SNE plot (Fig 3C) in the input and the latent space, where drugs with profiles divergent for different cell types in the input space (Fig 3C, left panel) could be observed to be encoded using very similar latent representations (Fig 3C, right panel). Similarly, a matrix containing correlations between latent

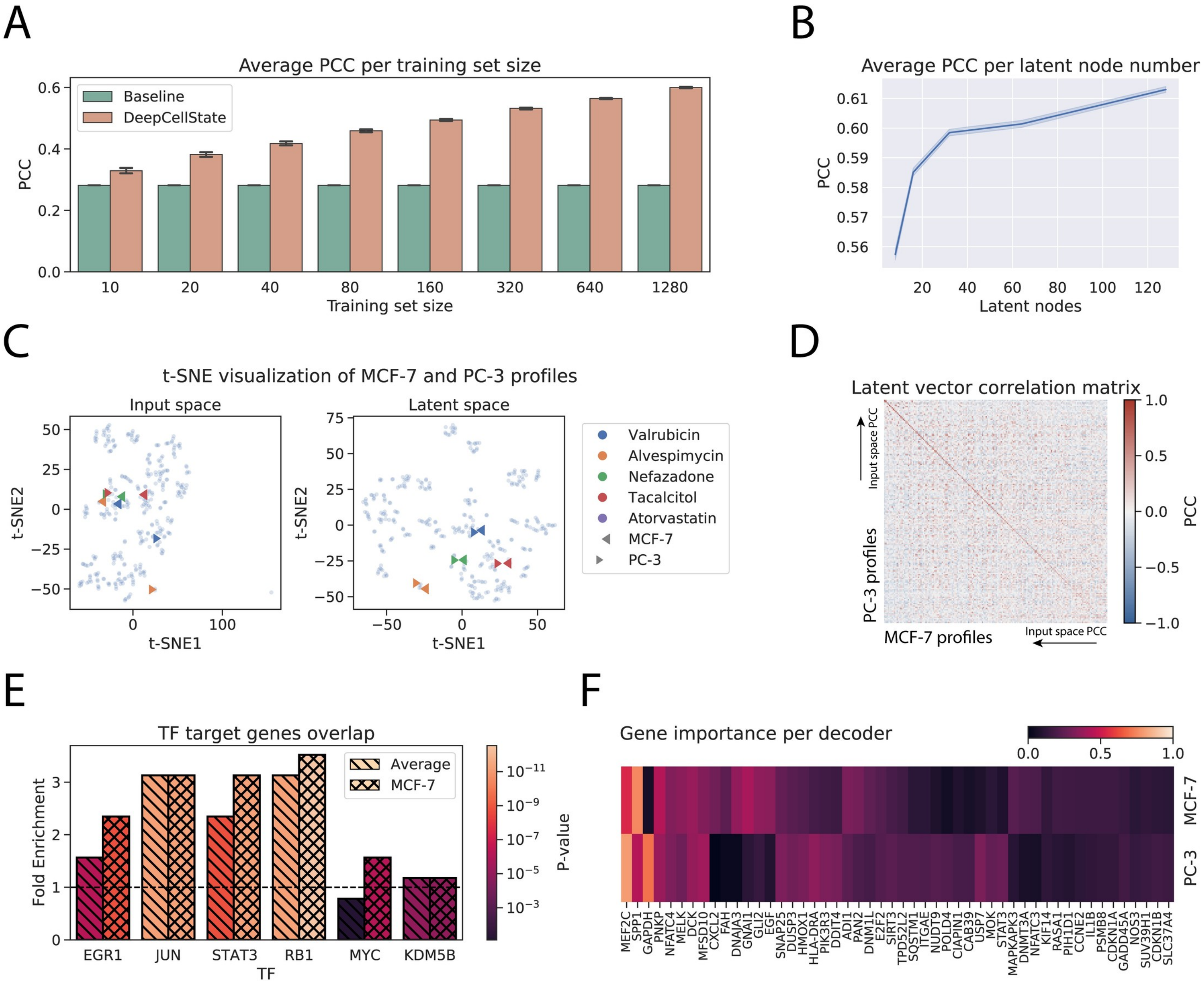

Fig 3. Analysis of the developed method. (**A**): Average PCC obtained for different size subsets of the training set. The y-axis indicates average correlation, and the x-axis indicates number of drugs used. (**B**): Dependency between the number of latent nodes and the average PCC. (**C**): t-SNE clustering of profiles in the input and the latent space. Profiles used are from test fold 1 of the two cell type model (MCF7, PC3) which achieved correlation above 0.40 PCC. (**D**): Matrix displaying latent vector response correlation. Drugs on each axis are sorted from high to low correlation in the input space. (**E**): Overlaps of top target genes of TFs based on chromatin immunoprecipitation data compared to target gene prediction based on DeepCellState trained model. (**F**): Heatmap of most important genes for MCF-7 and PC-3 decoders.

https://doi.org/10.1371/journal.pcbi.1009465.g003

responses for MCF-7 and PC-3 cells, sorted by the magnitude of the correlation of the input profiles, showed a consistently strong signal on the diagonal (Fig 3D), independent of the correlation between the responses in the input space.

It has been shown previously that autoencoders are able to capture transcription factor (TF) gene regulatory relationships [14,26]. To test if this was the case also for the DeepCellState framework, we used TF target gene lists from ChIP-Atlas [27] which include MCF-7 specific targets. After filtering, we computed the overlap between the targets implied by our model (see **Methods**) and targets based on ChIP-seq binding data, obtaining average fold enrichment of 2.5 (Fig 3E), with higher enrichment for the MCF-7 targets of four out of six transcription factors, compared to the general set of targets according to ChIP-Atlas. Identification of the most important genes in the cell type specific decoder response (see **Methods**) revealed enrichment of biological processes [28] that matched known properties of each cell type (Fig 3F, S7 Table). For the top genes common to the decoders, the high ranking hits were enriched for terms including "response to abiotic stimulus" (p = 0.0144) and "cellular response to chemical stimulus" (p = 0.018), expected since the model was trained using drug profiles (S8 Table). Additionally, several significantly enriched terms (p = 1.12e-04, p = 2.32e-04) were related to positive regulation of cell death and apoptotic process, likely due to some doses being toxic as previously observed [29]. Conversely, top genes of the MCF-7 decoder included several genes involved in mammary gland development (SRC, STAT5B, EGF). Similarly, BMP4, GLI2, and FGFR2, which are known to be involved in prostate morphogenesis, were among the top important genes of the PC-3 decoder. Taken together, these results suggest that the trained network is able to capture known biological processes and regulatory relationships, and that additional, deeper analysis may reveal relationships previously unnoticed.

Discussion

We here present DeepCellState, an autoencoder-based method that can successfully predict the transcriptional drug response in a cell type based on the response in another cell type. This problem was addressed in a recently proposed method that attempts to fill the gaps remaining in the combinatorial drug-cell space [2] through a computational framework that first arranges existing profiles into a three-dimensional array (or tensor) indexed by drugs, genes, and cell types, and then uses either local or global information to predict unmeasured profiles. Different approaches were evaluated, with the best one achieving an average correlation of 0.54 compared to 0.66 for DeepCellState. One issue with the proposed tensor based method is that the performance strongly depends on the density of the constructed tensor, meaning that cell type specific prediction of the response of a particular drug is heavily dependent on the amount of profiles for that same drug in other cell types. In contrast to the tensor-based methods, the DeepCellState framework is capable of predicting the response of drugs never seen previously by the network and can also make predictions that improve significantly over baseline using the response in completely unseen cell types as input. Our results on even a relatively small amount of LoF data suggest that the approach may be able to capture changes in cellular state in a more general way.

Previous autoencoder-based methods for analyzing transcriptome and other data capturing cellular states have mainly focused on predicting cell state at the single cell level [15,25,30]. Unlike these methods, DeepCellState can handle a large number of different perturbations, without requiring many data points for each. In a recent example, an autoencoder-based approach was used to translate between different domains of the single cell data [23], specifically between imaging and sequencing data. While the authors suggested the use of the approach for other types of translations, one requirement mentioned was that data come from the same cell population. With DeepCellState, we explicitly set out to predict the cell state in a particular cell type based on the state in one or more other cell types. Thus, we expect that the method can be readily applied to data sets measuring cell quantifiable characteristics other than transcriptome, e.g. imaging, Hi-C, or ATAC-seq data, with samples coming from different cell types.

It is a well-known fact that deep learning models are difficult to interpret. We have introduced several approaches to analyze the trained models. Using gradient descent optimization, it was possible to reveal cell type specific TF regulation captured by the network. By masking parts of the input profile, we were able to analyze each decoder, revealing the sets of genes which are used to make a prediction.

The results here serve as a proof-of-principle and may be improved by various modifications to the architecture and training strategy. Additionally, the performance may be increased by measuring a larger set of genes than the 978 "landmark genes" measured on the L1000 platform, or by adding measurements of additional transcriptional features such as non-coding RNAs and transcribed enhancers. Performance is also expected to increase as more data capturing additional cellular states becomes available, especially if produced under controlled and standardized experimental conditions.

Methods

LINCS data processing

Drug response expression profiles from LINCS phase 2 dataset (GSE70138) were used for training and evaluation. The latest level 5 data (signatures from aggregating replicates and converted to z-scores) was used (downloaded May 27, 2020). There are usually several treatment times and doses for each drug treatment; we averaged the profiles for the same drug across different timepoints and doses to reduce noise. Finally, the profiles were normalized to the range from -1 to +1 based on the whole data matrix.

Model architecture and training

The architecture of the autoencoder used in our method is shown in Fig 1B. The encoder consists of two dense layers with 512 and 256 neurons respectively. Dropout is applied to the input to increase the generalization capabilities of the model. The dropout rate is 0.5 which means half of the input genes will be set to 0 during training. This makes our model act as a denoising autoencoder since all values in the output layer need to be predicted. The last layer of the encoder is flattened and fed into a dense layer with 128 nodes, which represents our latent vector. The decoder takes the latent vector as an input and consists of two layers with 256 and 512 neurons respectively. The output layer has a direct connection to the dropout layer. The motivation is that some of the input can be "leaked" into the output [31].

We use L1 regularization for the latent layer, which acts as a sparsity constraint on the activity of the latent representations. Activation function is leaky relu [32] with alpha equal to 0.2 for all layers except the output layer, which uses tanh activation. The optimizer used is ADAM [33] with a learning rate equal to 1e-4 and batch size is 128. Validation set is employed for early stopping with patience of 5 epochs.

To initialize our model, we first train a regular autoencoder using profiles of all cell types as both input and output. Then we create a separate decoder for each cell type by copying the weights from the autoencoder. These decoders are trained one by one together with the shared encoder, where the inputs are profiles from all cell types and the outputs are profiles with the decoder's cell type only. In the final stage, when there has been no improvement in the validation set for 5 epochs, the encoder weights are frozen and only decoders are trained until there is no improvement. When making a prediction for a certain drug and cell combination, any cell profile can be used as an input. The cell specific decoder is used to make the final prediction.

DeepCellState is implemented in Python 3 using TensorFlow [34] library and the models were trained using NVIDIA Quadro RTX 6000 GPU. Training two cell decoders takes about one hour to complete, however new predictions can be made in less than a second.

External data processing

We trained an encoder and three decoders (HepG2, MCF-7 and PC-3) for external validation. The common drugs from the two external data sets were removed from the training and validation sets. The statin data set was generated by converting CAGE data to gene expression data by mapping to GENCODE version 34. CAGE tags in a 1000 base pair window around a landmark gene start were summed. We added a small value c = 2 to each gene value to avoid problems when calculating log fold change. Each generated profile was multiplied by a million and divided by the total sum of the profile. The resulting profiles for controls and treatments were averaged across replicates followed by a computation of log fold change after treatment for each gene. For the anticancer data we directly calculated log fold change between the average control and treatment values. In both statin and anticancer tests, we only used landmark genes, setting missing gene values to 0. Only the 24 h time point was used as it is the most common time point in LINCS phase 2 compound data (96% of the data).

Methods related to analysis/interpretation of model

Since it is hard to directly learn the inner-workings of the deep learning model, we study it by feeding in different inputs and analyzing the output. To calculate the top genes for each decoder, we iterated through all the data, for each profile using 100 random genes subsets of size 100 to input in the model. The gene sets which performed the best to make the prediction were considered to be important. For each profile we picked the top 10 subsets that performed the best. After doing this for all the profiles, we ranked the genes based on their frequency in these subsets.

For the TF analysis, we downloaded target genes for 22 TFs profiled in MCF-7 cells from ChIP-Atlas [27]. We filtered them by retaining TFs that were upregulated in at least one drug treatment profile (i.e. has value $>$ 0.5), and that had at least 50 targets in ChIP-Atlas (with targets defined as genes having a binding score of at least 100). This resulted in 6 remaining TFs and we picked the top 50 target genes for each. To make a prediction of target genes using DeepCellState, we started with a random profile and performed gradient descent optimization to find the input profile that maximizes the TF. As the obtained profile will also have other upregulated genes, the top 50 with the highest values are the DeepCellState prediction for the TF targets. For each TF we calculated fold enrichment and p-values using binomial statistic for the overlap between the DeepCellState set and the ChIP-Atlas set, using a similar method as described on the PANTHER website [28].

Comparison with alternative approaches

scGen [15] was designed to predict single cell perturbation responses based on VAE architecture, where an encoder *f* is used to learn the semantic embedding, *z*, of both the normal cell state and the stimulated state. The difference between such embeddings could then be considered as the embedding of the drug stimulation. When applying the method onto new cell types, the authors proposed to perform latent space vector arithmetic, adding the stimulation embedding, *δ*, to the normal cell embedding, and to pass the resulting vector through the decoder, *h*, obtaining the predicted responses.

Mathematically, we have

$$\delta_{n2s} = avg(z_{stimulated}) - avg(z_{normal}),$$

$$x'_{stimulated} = h(f(x_{normal}) + \delta_{n2s}),$$

where δ_{n2s} is the stimulation embedding; $z_{stimulated}$ is the embedding of the stimulated cell profile; z_{normal} is the embedding of the normal cell profile; x_{normal} is the input normal cell profile; $x'_{stimulated}$ is the predicted stimulated cell profile.

Regarding our task, since we have multiple drugs and the amount of data for each drug is limited, we modified the method to learn the difference between different cell types, instead of the difference between conditions. Consequently, we have

$$\delta_{p2m} = avg(z_{mcf7}) - avg(z_{pc3}),$$

$$x'_{mcf7} = h(f(x_{pc3}) + \delta_{p2m}).$$

Both the encoder, *f*, and decoder, *h*, are trained using the standard VAE training protocol described in the original paper [15] with our data as discussed above.

MAGAN [25] was developed to tackle the problem of aligning corresponding sets of samples. MAGAN is composed of two GANs, each with a generator network *G* that takes as input *X* and outputs a target dataset *X*′. For the minibatch (x_1, x_2), the loss for generator G_{12} is defined as sum of reconstruction loss, discriminator loss, and correspondence loss:

$$x_{12} = G_{12}(x_1),$$

$$x_{121} = G_{21}(x_{12}),$$

$$L_{G_1} = L_r + L_d + L_c,$$

$$L_r = L(x_1, x_{121}),$$

$$L_d = -\mathbb{E}_{x_1 \sim P_{x_1}}[\log D_2(x_{12})],$$

$$L_c = L(x_1, x_{12}).$$

The loss for discriminator D_1 is defined as:

$$L_d = -\mathbb{E}_{x_1 \sim P_{x_1}}[\log D_1(x_1) + \log D_1(x_{121})] - \mathbb{E}_{x_2 \sim P_{x_2}}[\log(1 - D_1(x_{21}))].$$

Since in our task all the data are paired, we used the following correspondence loss, as suggested in the original paper [25]:

$$L_c = MSE(x_{pc3 \to mcf7}, x_{mcf7}) + MSE(x_{mcf7 \to pc3}, x_{pc3}).$$

We used the total loss to train two generators, $G_{pc3 \to mcf7}$ and $G_{mcf7 \to pc3}$, together with two discriminators, D_{pc3} and D_{mcf7}.

To implement VAE with anchor loss, we have trained a standard VAE with two decoders and employed the following loss to ensure that all the *m* paired samples are encoded close to each other by encoder *E* in the latent space:

$$L_{anchor} = \sum_{i=1}^{m} |E(x_{i,pc3}) - E(x_{i,mcf7})|.$$

For DNPP and FaLRTC, we generated 10 test folds, selecting 10% of profiles randomly. When constructing the input for the methods, we set the test set profiles to “nan” values. The two methods then fill out the “nan” values and the new profiles are compared against the ground truth.

Supporting information

S1 Fig. DeepFake approach for the face swap. Similar facial expressions are encoded in a similar way in the latent space, while person specific facial details are reconstructed on the decoder side.
(TIF)

S2 Fig. Examples of MCF-7 profiles predicted with low, medium, and high PCC. (**A**): MCF-7 anastrozole response predicted with PCC 0.47. (**B**): MCF-7 GSK 2110183 response predicted with 0.61 PCC. (**C**): MCF-7 XL 888 response predicted with PCC 0.93.
(TIF)

S3 Fig. Different testing strategies of DeepCellState. (**A**): Results obtained by holding out entire drug families for the testing. (**B**): Performance of DeepCellSate by inclusion of profiles from other cell types in the training set. (**C**): Performance evaluation using completely unseen cell type as input. (**D**): Results for shRNA for LoF experiments profiles prediction.
(TIF)

S4 Fig. Comparison with DNPP and FaLRTC using 10-fold cross-validation.
(TIF)

S5 Fig. Comparison with DNPP and FaLRTC using the processed dataset.
(TIF)

S1 Table. Number of drugs in the ten drug families with the highest number of drugs.
(XLSX)

S2 Table. Performance of baseline and DeepCellState for each cell type.
(XLSX)

S3 Table. Performance of baseline and DeepCellState when predicting statin response profiles.
(XLSX)

S4 Table. Performance of baseline and DeepCellState when predicting response profiles of anticancer agents. DeepCellState* are results obtained after performing additional training on 14 anticancer agents and predicting for one repeated for each drug.
(XLSX)

S5 Table. Performance of DeepCellState and alternative methods for paired data modeling, using PC-3 responses to predict MCF-7 samples.
(XLSX)

S6 Table. Effect of regularization techniques on average PCC for different models.
(XLSX)

S7 Table. Top 50 genes identified for each decoder.
(XLSX)

S8 Table. Enriched Gene Ontology terms for top important genes in the latent layer.
(XLSX)

Acknowledgments

We would like to thank Andrew Tae-Jun Kwon and Bogumil Kaczkowski for insightful comments on the manuscript.

Author Contributions

Conceptualization: Erik Arner.

Formal analysis: Ramzan Umarov.

Investigation: Ramzan Umarov.

Methodology: Ramzan Umarov.

Project administration: Erik Arner.

Resources: Erik Arner.

Software: Ramzan Umarov.

Supervision: Erik Arner.

Validation: Ramzan Umarov, Yu Li.

Visualization: Ramzan Umarov.

Writing – original draft: Ramzan Umarov, Erik Arner.

Writing – review & editing: Ramzan Umarov, Yu Li, Erik Arner.

References

1. Niepel M, Hafner M, Duan Q, Wang Z, Paull EO, Chung M, et al. Common and cell-type specific responses to anti-cancer drugs revealed by high throughput transcript profiling. Nat Commun. 2017 30; 8(1):1186. https://doi.org/10.1038/s41467-017-01383-w PMID: 29084964
2. Hodos R, Zhang P, Lee H-C, Duan Q, Wang Z, Clark NR, et al. Cell-specific prediction and application of drug-induced gene expression profiles. Pac Symp Biocomput Pac Symp Biocomput. 2018; 23:32–43. PMID: 29218867
3. Iorio F, Bosotti R, Scacheri E, Belcastro V, Mithbaokar P, Ferriero R, et al. Discovery of drug mode of action and drug repositioning from transcriptional responses. Proc Natl Acad Sci U S A. 2010 Aug 17; 107(33):14621–6. https://doi.org/10.1073/pnas.1000138107 PMID: 20679242
4. Iorio F, Rittman T, Ge H, Menden M, Saez-Rodriguez J. Transcriptional data: a new gateway to drug repositioning? Drug Discov Today. 2013 Apr; 18(7–8):350–7. https://doi.org/10.1016/j.drudis.2012.07.014 PMID: 22897878
5. Ye C, Ho DJ, Neri M, Yang C, Kulkarni T, Randhawa R, et al. DRUG-seq for miniaturized high-throughput transcriptome profiling in drug discovery. Nat Commun. 2018 17; 9(1):4307. https://doi.org/10.1038/s41467-018-06500-x PMID: 30333485
6. Subramanian A, Narayan R, Corsello SM, Peck DD, Natoli TE, Lu X, et al. A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles. Cell. 2017 Nov 30; 171(6):1437–1452.e17. https://doi.org/10.1016/j.cell.2017.10.049 PMID: 29195078
7. Lamb J, Crawford ED, Peck D, Modell JW, Blat IC, Wrobel MJ, et al. The Connectivity Map: using gene-expression signatures to connect small molecules, genes, and disease. Science. 2006 Sep 29; 313 (5795):1929–35. https://doi.org/10.1126/science.1132939 PMID: 17008526
8. Eraslan G, Simon LM, Mircea M, Mueller NS, Theis FJ. Single-cell RNA-seq denoising using a deep count autoencoder. Nat Commun. 2019 Dec; 10(1):390. https://doi.org/10.1038/s41467-018-07931-2 PMID: 30674886
9. Talwar D, Mongia A, Sengupta D, Majumdar A. AutoImpute: Autoencoder based imputation of single-cell RNA-seq data. Sci Rep. 2018 Dec; 8(1):16329. https://doi.org/10.1038/s41598-018-34688-x PMID: 30397240
10. Danaee P, Ghaeini R, Hendrix DA. A DEEP LEARNING APPROACH FOR CANCER DETECTION AND RELEVANT GENE IDENTIFICATION. Pac Symp Biocomput Pac Symp Biocomput. 2017; 22:219–29. https://doi.org/10.1142/9789813207813_0022 PMID: 27896977
11. Tan J, Hammond JH, Hogan DA, Greene CS. ADAGE-Based Integration of Publicly Available Pseudomonas aeruginosa Gene Expression Data with Denoising Autoencoders Illuminates Microbe-Host Interactions. Gilbert JA, editor. mSystems. 2016 Feb 25; 1(1):mSystems.00025–15, e00025-15. https://doi.org/10.1128/mSystems.00025-15 PMID: 27822512

12. Xie R, Wen J, Quitadamo A, Cheng J, Shi X. A deep auto-encoder model for gene expression prediction. BMC Genomics. 2017 Nov; 18(S9):845. https://doi.org/10.1186/s12864-017-4226-0 PMID: 29219072

13. Rampášek L, Hidru D, Smirnov P, Haibe-Kains B, Goldenberg A. Dr.VAE: improving drug response prediction via modeling of drug perturbation effects. Bioinforma Oxf Engl. 2019 01; 35(19):3743–51. https://doi.org/10.1093/bioinformatics/btz158 PMID: 30850846

14. Chen L, Cai C, Chen V, Lu X. Learning a hierarchical representation of the yeast transcriptomic machinery using an autoencoder model. BMC Bioinformatics. 2016 Dec; 17(S1):S9. https://doi.org/10.1186/s12859-015-0852-1 PMID: 26818848

15. Lotfollahi M, Wolf FA, Theis FJ. scGen predicts single-cell perturbation responses. Nat Methods. 2019 Aug; 16(8):715–21. https://doi.org/10.1038/s41592-019-0494-8 PMID: 31363220

16. Nguyen TT, Nguyen CM, Nguyen DT, Nguyen DT, Nahavandi S. Deep learning for deepfakes creation and detection. ArXiv Prepr ArXiv190911573. 2019;1.

17. Tariyal S, Majumdar A, Singh R, Vatsa M. Deep Dictionary Learning. IEEE Access. 2016; 4:10096–109.

18. Salakhutdinov R, Hinton G. Deep Boltzmann Machines. In: Dyk D van, Welling M, editors. Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics [Internet]. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR; 2009. p. 448–55. (Proceedings of Machine Learning Research; vol. 5). Available from: http://proceedings.mlr.press/v5/salakhutdinov09a.html

19. Vincent P, Larochelle H, Bengio Y, Manzagol P-A. Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning. 2008. p. 1096–103.

20. Huang C-T, Hsieh C-H, Chung Y-H, Oyang Y-J, Huang H-C, Juan H-F. Perturbational Gene-Expression Signatures for Combinatorial Drug Discovery. iScience. 2019 May 31; 15:291–306. https://doi.org/10.1016/j.isci.2019.04.039 PMID: 31102995

21. Huang Y, Furuno M, Arakawa T, Takizawa S, de Hoon M, Suzuki H, et al. A framework for identification of on- and off-target transcriptional responses to drug treatment. Sci Rep. 2019 26; 9(1):17603. https://doi.org/10.1038/s41598-019-54180-4 PMID: 31772269

22. Monks A, Zhao Y, Hose C, Hamed H, Krushkal J, Fang J, et al. The NCI Transcriptional Pharmacodynamics Workbench: A Tool to Examine Dynamic Expression Profiling of Therapeutic Response in the NCI-60 Cell Line Panel. Cancer Res. 2018 15; 78(24):6807–17. https://doi.org/10.1158/0008-5472.CAN-18-0989 PMID: 30355619

23. Yang KD, Belyaeva A, Venkatachalapathy S, Damodaran K, Katcoff A, Radhakrishnan A, et al. Multi-domain translation between single-cell imaging and sequencing data using autoencoders. Nat Commun. 2021 Dec; 12(1):31. https://doi.org/10.1038/s41467-020-20249-2 PMID: 33397893

24. Isola P, Zhu J-Y, Zhou T, Efros AA. Image-to-Image Translation with Conditional Adversarial Networks. ArXiv161107004 Cs [Internet]. 2018 Nov 26 [cited 2021 Feb 8]; Available from: http://arxiv.org/abs/1611.07004

25. Amodio M, Krishnaswamy S. MAGAN: Aligning Biological Manifolds. ArXiv180300385 Cs [Internet]. 2018 Feb 9 [cited 2021 Jan 21]; Available from: http://arxiv.org/abs/1803.00385

26. Tan J, Doing G, Lewis KA, Price CE, Chen KM, Cady KC, et al. Unsupervised Extraction of Stable Expression Signatures from Public Compendia with an Ensemble of Neural Networks. Cell Syst. 2017 26; 5(1):63–71.e6. https://doi.org/10.1016/j.cels.2017.06.003 PMID: 28711280

27. Oki S, Ohta T, Shioi G, Hatanaka H, Ogasawara O, Okuda Y, et al. ChIP-Atlas: a data-mining suite powered by full integration of public ChIP-seq data. EMBO Rep. 2018; 19(12). https://doi.org/10.15252/embr.201846255 PMID: 30413482

28. Mi H, Muruganujan A, Ebert D, Huang X, Thomas PD. PANTHER version 14: more genomes, a new PANTHER GO-slim and improvements in enrichment analysis tools. Nucleic Acids Res. 2019 08; 47(D1):D419–26. https://doi.org/10.1093/nar/gky1038 PMID: 30407594

29. Sirci F, Napolitano F, Pisonero-Vaquero S, Carrella D, Medina DL, di Bernardo D. Comparing structural and transcriptional drug networks reveals signatures of drug activity and toxicity in transcriptional responses. NPJ Syst Biol Appl. 2017; 3:23. https://doi.org/10.1038/s41540-017-0022-3 PMID: 28861278

30. Amodio M, van Dijk D, Srinivasan K, Chen WS, Mohsen H, Moon KR, et al. Exploring single-cell data with deep multitasking neural networks. Nat Methods. 2019 Nov; 16(11):1139–45. https://doi.org/10.1038/s41592-019-0576-7 PMID: 31591579

31. Mirsky Y, Lee W. The Creation and Detection of Deepfakes: A Survey. ArXiv200411138 Cs Eess [Internet]. 2020 Sep 13 [cited 2020 Dec 12]; Available from: http://arxiv.org/abs/2004.11138

32. Maas AL, Hannun AY, Ng AY. Rectifier nonlinearities improve neural network acoustic models. In: Proc icml. 2013. p. 3.

33. Kingma DP, Ba J. Adam: A method for stochastic optimization. ArXiv Prepr ArXiv14126980. 2014;

34. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. Tensorflow: A system for large-scale machine learning. In: 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16). 2016. p. 265–83.