Interpretable Machine Learning for Perturbation Biology

Bo Yuan[1,2,3*#], Ciyue Shen[1,2,3*#], Augustin Luna[1,2,3], Anil Korkut[4], Debora S. Marks[3,5], John Ingraham[6], Chris Sander[1,2,3#]

1 Dept. of Cell Biology, Harvard Medical School

2 cBio Center, Dept. of Data Sciences, Dana-Farber Cancer Institute

3 Broad Institute of MIT and Harvard

4 Dept. of Bioinformatics & Computational Biology, The University of Texas MD Anderson Cancer Center

5 Dept. of Systems Biology, Harvard Medical School

6 MIT Computer Science & Artificial Intelligence Lab

* Joint first authors

Correspondence to mathcellbox@gmail.com reaches the principal authors

Abstract

Systematic perturbation of cells followed by comprehensive measurements of molecular and phenotypic responses provides an informative data resource for constructing computational models of cell biology. Models that generalize well beyond training data can be used to identify combinatorial perturbations of potential therapeutic interest. Major challenges for machine learning on large biological datasets are to find global optima in an enormously complex multi-dimensional solution space and to mechanistically interpret the solutions. To address these challenges, we introduce a hybrid approach that combines explicit mathematical models of dynamic cell biological processes with a machine learning framework, implemented in Tensorflow. We tested the modeling framework on a perturbation-response dataset for a melanoma cell line after drug treatments. The models can be efficiently trained to accurately describe cellular behavior, as tested by cross-validation. Even though completely data-driven and independent of prior knowledge, the resulting *de novo* network models recapitulate some known interactions. The main predictive application is the identification of combinatorial candidates for cancer therapy. The approach is readily applicable to a wide range of kinetic models of cell biology.

Introduction

The emergence of resistance to single anticancer agents has highlighted the importance of developing combinations of agents as a more robust therapeutic approach to cancer treatment (Fitzgerald *et al.*, 2006; Garraway and Jänne, 2012; Mansoori *et al.*, 2017; Mokhtari *et al.*, 2017). However, experimental screening of all possible pairwise or higher-order combinations of currently available agents is practically unrealistic. The space of potential new therapeutic targets is even larger and more challenging to explore experimentally. To efficiently narrow down the search space and nominate promising sets of experimentally testable candidates, computational models have been used to predict cellular responses based on sets of perturbation experiments (Azmi *et al.*, 2010; Ryall and Tan, 2015; Cheng, Kovács and Barabási, 2019), but these have been limited in scope. The ability to model cell biology at a larger scale and to infer causal mechanisms to generalize to unobserved perturbations is critical in facilitating the search for combinatorial, potentially therapeutic candidates.

Perturbation-response profiling in cell biology

In order to understand cell behavior, a large variety of experimental approaches has been used to profile cellular responses under different perturbations. Biochemical and cell biological experiments testing relationships of particular protein-protein pairs have, for many years, been successfully used to identify signaling cascades (Cerami *et al.*, 2011; Wrzodek *et al.*, 2013; Croft *et al.*, 2014), but one-by-one perturbation experiments are laborious and the resulting models, while insightfully descriptive, typically do not provide the ability to quantitatively predicting detailed molecular or system-level responses. Phenotypic screening collects high-throughput information on whole-cell responses with univariate readouts such as cell viability or growth rate (Lamb *et al.*, 2006; Cheung *et al.*, 2011; Wang *et al.*, 2014; McDonald *et al.*, 2017; Tsherniak *et al.*, 2017). In order to resolve intracellular interactions and provide mechanistic insights,

systematic methods have been developed to profile post-perturbational molecular responses with increasing coverage, e.g., changes in transcript (Dixit *et al.*, 2016; Niepel *et al.*, 2017; Norman *et al.*, 2019) and protein (Korkut *et al.*, 2015; Hill *et al.*, 2017) levels. These rich datasets challenge computational methods to describe mechanisms more comprehensively and to quantitatively model cell responses to unseen perturbations, e.g., for the design of experiments to test mechanistic hypotheses or for the design of combinatorial therapeutic interventions.

Computational modeling

Various computational methods have been developed to infer interactions and predict cellular responses (Vanhaelen, Aliper and Zhavoronkov, 2017). Static models use, e.g., co-expression models (Carter *et al.*, 2004; Wang *et al.*, 2009; Babur *et al.*, 2010), maximum entropy networks (Lezon *et al.*, 2006; Locasale and Wolf-Yadlin, 2009), or mutual information related methods (Meyer, Lafitte and Bontempi, 2008; Chan, Stumpf and Babtie, 2017), to construct network models of molecular interactions (Şenbabaoğlu *et al.*, 2016; Yi *et al.*, 2017), or use regression models to directly predict cellular responses based on molecular perturbation-response measurements (Dixit *et al.*, 2016; Norman *et al.*, 2019). On the other hand, dynamic models, such as Boolean network models (D'haeseleer, Liang and Somogyi, 2000), fuzzy logic models (Aldridge *et al.*, 2009), dynamic Bayesian networks (Zou and Conzen, 2005) and ordinary differential equation (ODE) network models (Nyman *et al.*, 2019; Gardner *et al.*, 2003), can provide mechanistic insight in terms of propagation of cellular signals to phenotypic response over time, but typically require prior knowledge of interaction parameters and thus currently only work for small systems (Gardner *et al.*, 2003; Klinger *et al.*, 2013). New algorithms to parametrize large-scale mechanistic models, while computationally efficient, require prior knowledge of a set of relevant interactions (Fröhlich *et al.*, 2018). Insufficient prior knowledge is one of the major constraints for modeling large systems, e.g., in that prior information is not available for all components or is aggregated from disparate experimental sources and thus lacks uniform context. A more rigorous approach is to use uniform datasets generated in systematic experiments in one

experimental context and then perform de novo structure inference of an interaction network specific for that context. Given such data for large systems, the computational challenge is to search for optimal interaction parameter sets in a complex multi-dimensional solution space. Previous dynamic optimization approaches such as Monte Carlo（MC）methods and belief propagation (BP) algorithms, have been used to construct data-driven network models (Bruggeman *et al.*, 2002; Nelander *et al.*, 2008; Hug *et al.*, 2013; Klinger *et al.*, 2013; Korkut *et al.*, 2015). Still, these may not efficiently scale to larger systems (e.g., MC) or may require excessive approximations for the chosen mathematical model to facilitate efficient exploration of solution space (e.g., independent row approximation in BP) (Bruggeman *et al.*, 2002; Korkut *et al.*, 2015). Therefore, to achieve good accuracy of parameter inference for larger systems and to gain the ability to generalize to more sophisticated kinetic models, a more general and potentially more powerful data-driven modeling framework is needed.

Machine learning and interpretability

Recently, deep learning has become an effective data-driven framework capable of generating predictions for large and complex systems. Gradient descent implemented with automatic differentiation, which has been broadly used in training graphical models, allows efficient parameter optimization in complex network systems. This framework has been successfully applied to many domains of biomedical research, from pathology image classification (Hou *et al.*, 2016; Esteva *et al.*, 2017) to sequence motif detection (Zhou and Troyanskaya, 2015). While predictive power of deep learning models is often impressive, their interpretation, which is crucial for providing understandable and, therefore, more trustable predictions, remains challenging. The complex multi-layer network architecture of most deep learning models lacks explicit representations and consequent direct interpretation. This difficulty is sometimes called the “black box” problem (Montavon, Samek and Müller, 2018). To address this problem, we apply a deep learning optimization approach to learn a data-driven model (called “CellBox”) that incorporates an explicitly interpretable network of interactions between cellular

components, instead of a black-box neural network, while aiming to maintain a high level of learning performance.

CellBox is designed to be a framework for computational modeling of cellular responses to perturbations that i) links perturbations to molecular and phenotypic changes in a unified computational model; ii) quantifies time-dependent (dynamic) cellular responses; iii) promises training efficiency and scalability for large-scale systems; iv) is interpretable in terms of interactions that can be compared to established models of molecular biology, such as signaling pathways. Here, we construct a non-linear ordinary differential equation (ODE) based model representing a biological network of 99 components connecting perturbations, protein response, and phenotypes to simulate dynamic cellular behavior. The network connections are directly learned from post-perturbational data under 89 experimental conditions with the objective of accurately reproducing the molecular and cellular responses on training data. To reach this objective, we implemented gradient descent with automatic differentiation to infer interaction parameters in the ODE network, which can then be exposed to novel perturbations to predict cell behavior. The key performance criterion for the data-driven model trained with a relatively small set of experiments is whether the model is able to provide reasonably accurate predictions on a large set of unseen perturbation conditions. Anticipating the availability of increasingly informative perturbation-response data sets in diverse areas of cell biology, we present CellBox as a generally applicable framework for modeling a broad range of dynamic cell behavior.

Results

CellBox model of perturbation biology

In order to construct a data-driven model to predict the dynamics of molecular and cellular behavior under combinations of drug treatments, the perturbation data has to have 1) paired measurements of changes in protein levels and cellular behavior for a set of perturbations; and 2) training and withheld data to test model performance. Here, we used a perturbation dataset for the melanoma cell line SK-Mel-133 (Korkut *et al.*, 2015), which contains molecular and phenotypic response profiles of cells treated with 12 different drugs and their pairwise combinations (Figure 1a, S1). For each of the 89 perturbation conditions, levels of 82 selected proteins and phosphoproteins were measured in cell lysates before and 24 hours after perturbation on antibody-based Reverse Phase Protein Arrays (RPPA). In parallel, cellular phenotypes were assayed, including cell cycle progression and cell viability. With parallel measurements of proteomic and phenotypic responses to a systematic set of perturbations, this dataset provides sufficient information to construct network models that quantitatively link molecular changes to cellular responses.

We used a set of ordinary differential equations (ODEs with a non-linear envelope) (Figure 1b) to model the dynamic responses of the system to drug perturbations (See Methods). The parameters of the ODEs (w_{ij}, ~10,000 in total) are the interaction strengths between the entities in the network model. The simplicity of the interaction dynamics (Figure 1b), the non-linear envelope, as well as the restoration term $-\alpha_i x_i(t)$ are computational devices, roughly analogous to mean-field approaches, to account for the fact that the data is limited to a relatively small fraction of all cellular components and to avoid instabilities (Nelander *et al.*, 2008; Molinelli *et al.*, 2013; Korkut *et al.*, 2015). The interaction parameters were randomly initialized and updated throughout the model training process, with the objective to minimize a loss function. For the loss

function, we chose the Euclidean distance between experimental data and the results of the numerical simulation of the ODE model, plus an L1 regularization penalty on network density to avoid overfitting (See Methods, eqn. 3). We used Heun's ODE solver (Süli and Mayers, 2003) to numerically simulate the ODE system and the Adam optimizer (Kingma and Ba, 2014) with automatic differentiation to minimize the loss function. Taken together, we constructed an ODE model of a cell biological system trained using perturbation data, which we named CellBox.

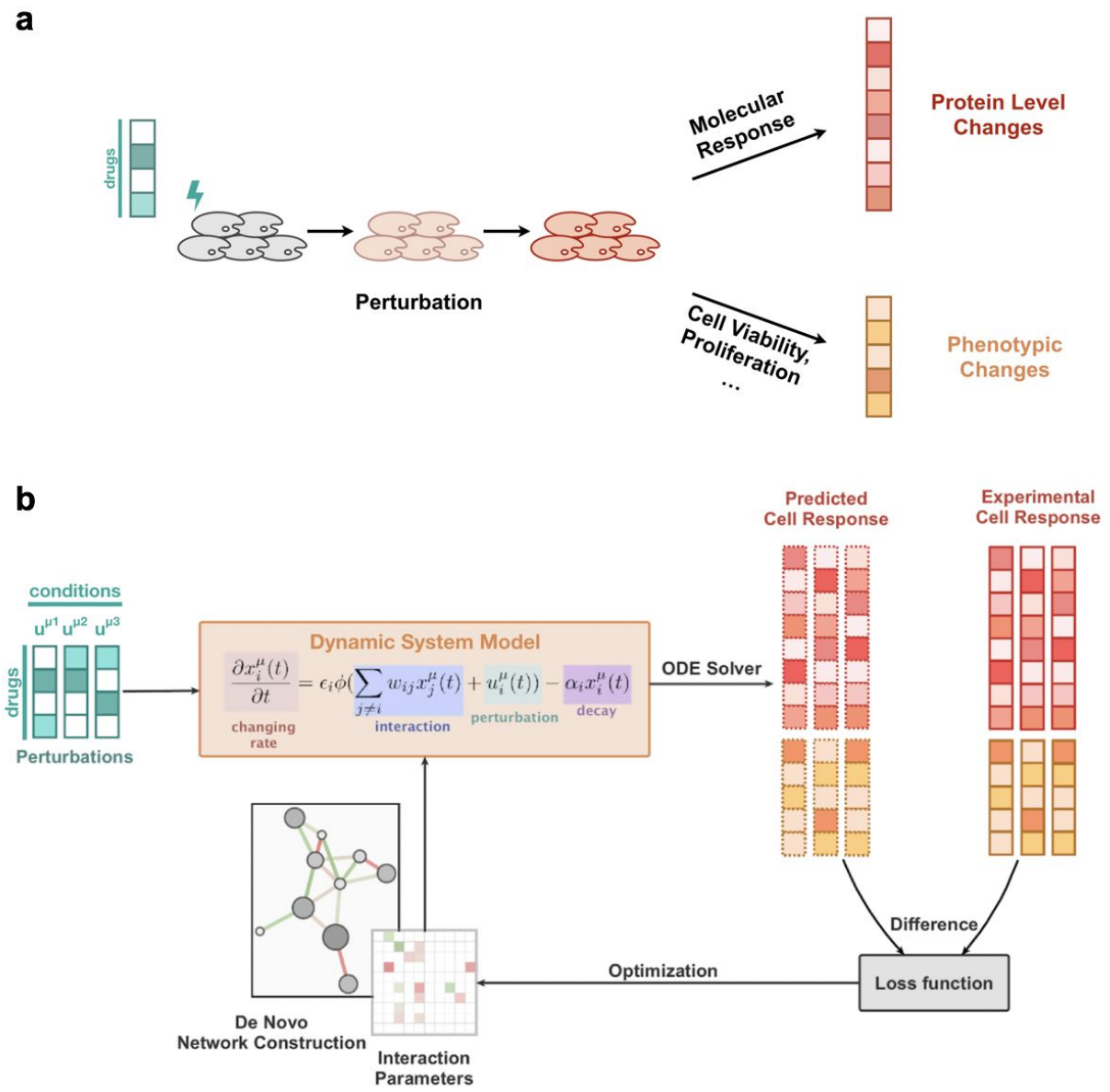


Figure 1. CellBox: dynamic modeling of cellular systems with perturbation data.

a. Perturbations such as drugs are used to disturb the cellular system. The cell responses, including protein and phosphoprotein level changes, and phenotypic changes, were measured to provide information for model construction. **b.** Systematic responses of the cells under various drug perturbations were used to construct an interpretable machine learning model. CellBox models system behavior in terms of interaction parameters connecting molecular (proteins and phosphoproteins) and phenotypic variables using a set of differential equations. CellBox was trained iteratively by optimizing interaction parameters to fit the numerically simulated system response to experimental observations. After training on pairwise data of input perturbation and output system behaviors, the CellBox model can be used to predict the cellular response to arbitrary perturbation conditions.

CellBox can be trained on perturbation data to accurately predict cell response.

In order to test the prediction performance of this training scheme, we randomly selected 70% of the perturbation data (n = 62 conditions) for training and withheld the rest 30% (n = 27 conditions) for testing. 20% of the training data was used as a validation set to stop model training when the performance on the validation set did not further improve. We manually fine-tuned the hyperparameters, including learning rate, regularization, and ODE simulation time, to increase the training efficiency (Figure S2). At the end of the training, the numerical solutions of the ODE model converged efficiently to experimental data (Figure 2a, 2b). We repeated the modeling scheme with 1,000 independent random data partitions to construct models for each partition. The average predictions on test sets across all models and all conditions correlate with experimental data with a Pearson's correlation coefficient of 0.94 (Figure 2c). A more refined analysis of individual perturbation conditions showed that the model trains equally well for all conditions (Figure 2d), and that model performance is independent of data scaling normally applied to protein profiling data (Figure S3). The results illustrate that the CellBox model can be efficiently trained with perturbation data to predict cell response to experimentally applied perturbations accurately.

Even though ~70% of the models reached steady solutions of the ODEs (Figure 2b), some models converged to oscillatory solutions (Figure S4a). In order to test whether the oscillation is an artifact of data partitioning during model training, we re-trained the models with the same train-test data partitioning but multiple different random seeds for the computational optimizer (See Methods). For each partition of the training data, both steady and oscillatory solutions can result (Figure S4a-d), and the emergence of oscillatory solutions is independent of ODE solvers (Figure S4e-f). Based on the assumption that the population average of cell response reaches a stable and non-oscillating steady state 24 hours after drug treatment, we excluded the oscillatory models in the following analysis (See Methods). Taken together, these results indicate that CellBox, a data-driven ODE-based cellular system model, can be trained to

accurately predict dynamics of cell response, without any requirement of prior knowledge about the relationship between particular protein levels or phenotypes.

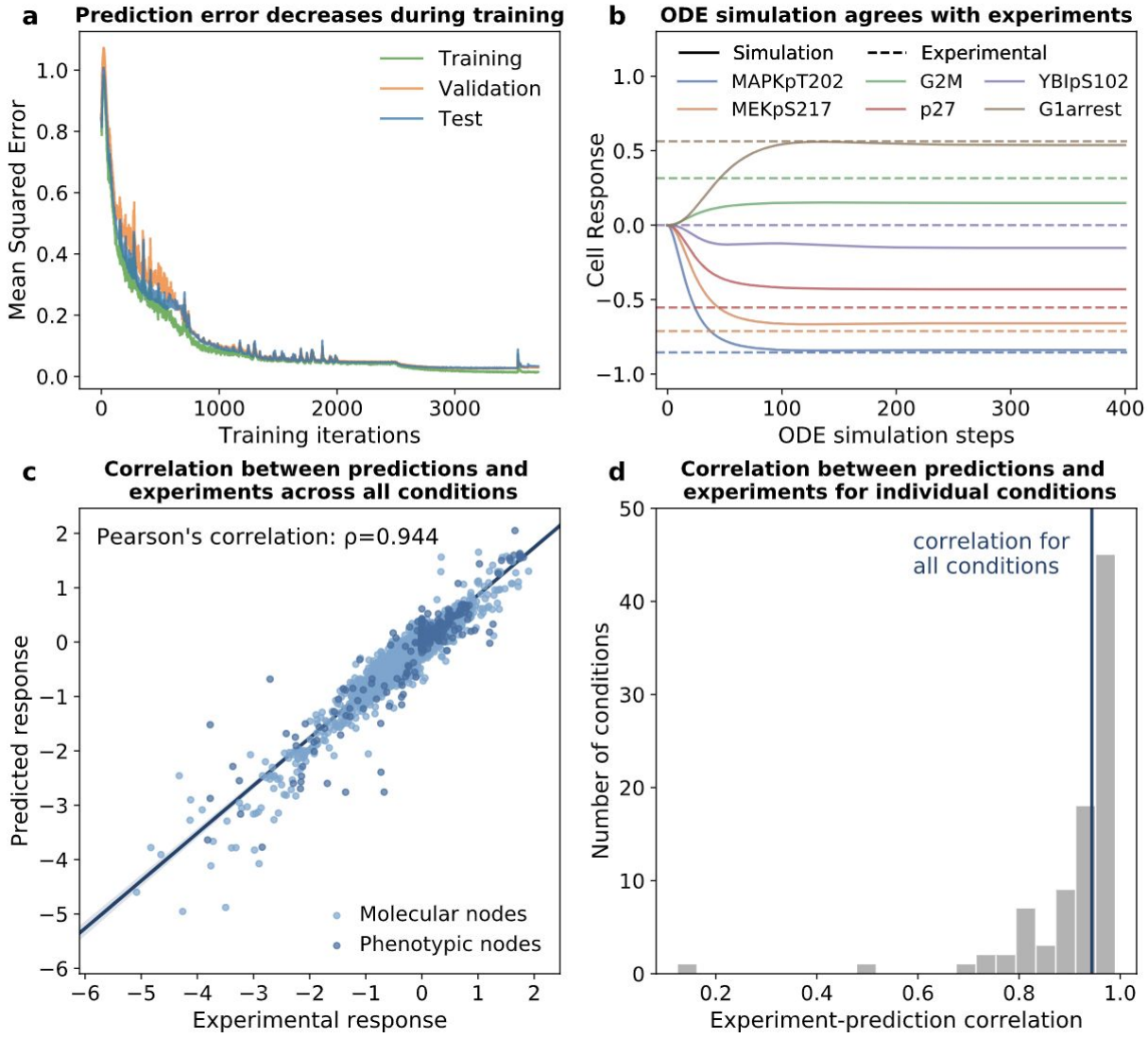


Figure 2. CellBox convergence and prediction accuracy on randomly partitioned training-test datasets.

a. Over training iterations, the mean squared error on the training set (56% of the entire dataset), validation set (14%) and test set (30%) decreased nearly monotonically, and the models converged at the end of the training. **b.** The predicted molecular and phenotypic responses at the steady state of the ODE simulations agree with the

experimental data on the test set. A subset of molecular measurements (MAPKpT202, YB1pS102, MEKpS217, and p27) and phenotypic measurements (G2M and G1arrest) are shown. Cell response is defined as the $\log_2$ ratio of post- and pre-perturbation measurements. The annotations and the full set of measurements are in Supplementary Table S1. **c.** Across 1,000 models trained with different data partitions, the average predicted responses correlate with experimental observations (Pearson's correlation ρ = 0.944, regression line in dark blue with 95% confidence interval). Each point represents one measurement, either molecular or phenotypic, in one perturbation condition. **d.** Nearly all predictions for individual conditions have high correlations with experimental measurements.

CellBox model predicts cell response for single-to-combo and leave-one-drug-out cross-validations.

Even though the model makes accurate predictions with different training data, data partitioning, especially random partitioning, raises the concern of information sharing between training and test datasets. Combinatorial conditions in both datasets might share the same drugs such that the test set might not be truly independent of training and, therefore, is suboptimal for rigorous evaluation of the model performance. Moreover, the ability to predict the combinatorial effect of a drug, e.g., dominant, additive, synergistic, when none of its combinations has been seen by the model, is a non-trivial challenge in the context of making accurate predictions of experimentally untested drug combinations.

In order to address these points, rather than training the model with random data partitioning, we instead designed more rigorous tasks: single-to-combo (Figure 3a, S7) and leave-one-drug-out cross-validation (Figure 3b, 3c) for each drug. In the single-to-combo analysis, all single-drug treatment conditions were used for training, and predictions were made on all combinatorial drug conditions. In leave-one-drug-out cross-validation, all the combination conditions containing the treatment of a particular

drug with or without the corresponding single drug conditions were withheld while the rest of the conditions were used for training. In these more stringent tests, we found that the predicted values for withheld data were still highly correlated with the experimental observations (average Pearson's correlation: 0.93 for single-to-combo; 0.94 for leave-one-drug-out with single conditions, similar to that of the training with random partition; 0.79 for complete leave-one-drug-out). Under all three scenarios, on this dataset, CellBox outperforms the belief propagation (BP) dynamic model approach previously used in perturbation biology (Korkut *et al.*, 2015) in terms of predictive accuracy. These results indicate that the CellBox model can be trained with a relatively small set of perturbation data and that its predictions can be generalized to unseen combinatorial perturbations. In particular, CellBox models predict more accurately than linear models in the single-to-combo scenario (Figure S7), suggesting that CellBox can capture the non-additive (synergistic or antagonistic) effects, which is particularly useful in nominating therapeutic drug combinations.

CellBox models are dynamic network models of a cell biological system. To test whether such interpretable network models of molecular interactions help increase model predictive power, we compared the results to those of a static biological network model and a deep neural network model. The static network model was constructed by learning co-expression correlation for each pair of protein nodes (Co-exp) while the deep neural network model was trained to directly regress phenotypic changes against parameterized perturbations (NN) (see Methods). In all three tasks, the static network models had lower accuracy relative to the dynamic CellBox. The NN had comparable performance to CellBox in the cross-validation for individual drugs, but its performance dropped significantly in the single-to-combo analysis (Figure 3a). Furthermore, the NN was unable to generalize to unseen targets whose information is completely excluded from training (Figure 3c, Figure S6). Taken together, due to the lack of mechanistic and dynamic information, static network or direct regression models appear to be less suitable for facilitating the search for combinatorial targets.

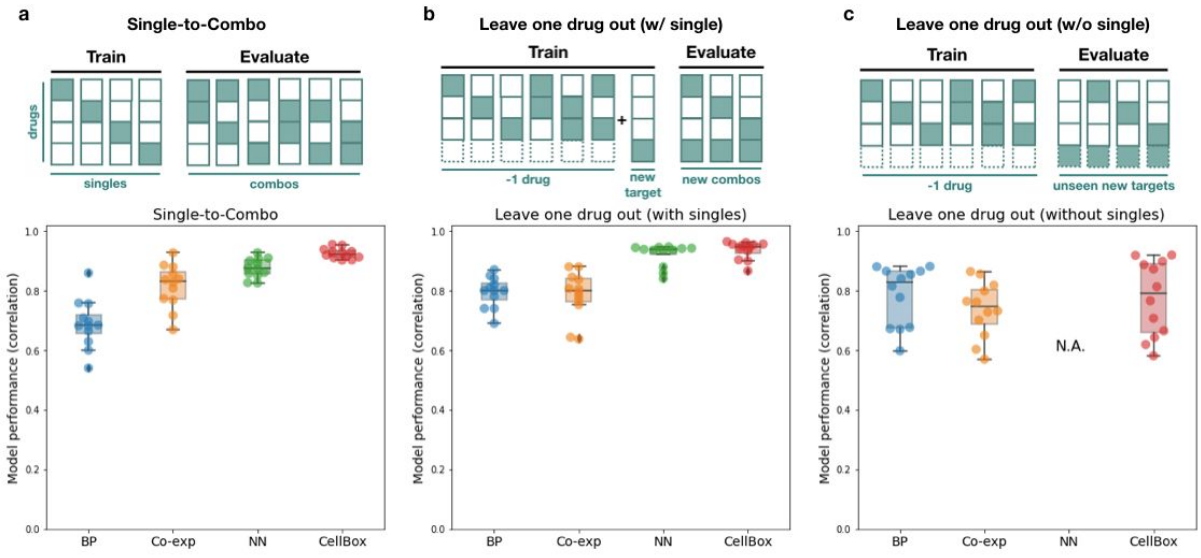


Figure 3. CellBox can accurately predict cell response for single-to-combo and leave-one-drug-out cross-validations.

a. When only single conditions were used for training (single-to-combo), the CellBox models predict the effects of combinatorial conditions with high accuracy and outperform the dynamic network model inferred using belief propagation (BP), the static co-expression network model (Co-exp), and a neural network regression model (NN) trained on the same data. **b.** When combinatorial conditions associated with one drug were withheld from training, the CellBox models retain high accuracy for predicting the effects of unseen drug pairs. **c.** When all conditions associated with one drug were withheld from training, the ODE network models predict the effects of the withheld drug with reduced accuracy, but direct regression models such as NN cannot generalize to unseen targets at all. For each model type, the performance was evaluated by Pearson's correlation between predicted cell response and experimental cell response.

Model performance is robust against noise and reduced training set size.

To examine model robustness of the CellBox models against a reduction in training data, we tested the stability of model performance when the data quality or quantity is compromised. To test the former, we introduced different levels of multiplicative Gaussian noise (see Methods) into the input molecular and cellular response data and trained models on the resultant noisy datasets. The assumption behind such multiplicative noise is that the uncertainty and noise in experimental measurements arise around the true values. When comparing the predicted response in test sets to the experimental data, we found that the predictions from training on the noisy data retain similarly high correlations to experimental data as those trained on the original data, even with the addition of 5% multiplicative Gaussian noise (Figure 4a). As the magnitude of the noise increases, the model performance decreases gradually in terms of both convergence (Figure S8a) and predictive power (Figure 4a). We observed similar behavior when challenging the CellBox models with additive Gaussian noise (Figure 4b, S8b). The predictivity of CellBox models can tolerate an addition of up to $\sigma_{add} = 0.20$ additive Gaussian noise, i.e., about half the data standard deviation $\sigma_{data} = 0.46$. We conclude that model performance is stable in the presence of moderate experimental errors.

To test the dependency of model performance on data quantity, we trained the model on subsamples of the experimental dataset. We trained models with varying amounts of data (from 10% to 90% in steps of 10%) and found that the models could make accurate predictions of withheld data with as little as 40% of the complete dataset (Figure 4c). We found that increasing the size of the training set further has diminishing returns in terms of model performance. This implies, on the dataset used here with an interaction network of ~100 components, that a comparatively small number of perturbation conditions (40-100, rather than directly testing all ~3,000 possible combinations) are sufficient for constructing reasonably predictive models. This

example may be a useful guide for power calculations for systems with hundreds of measured components, which would be of considerable interest.

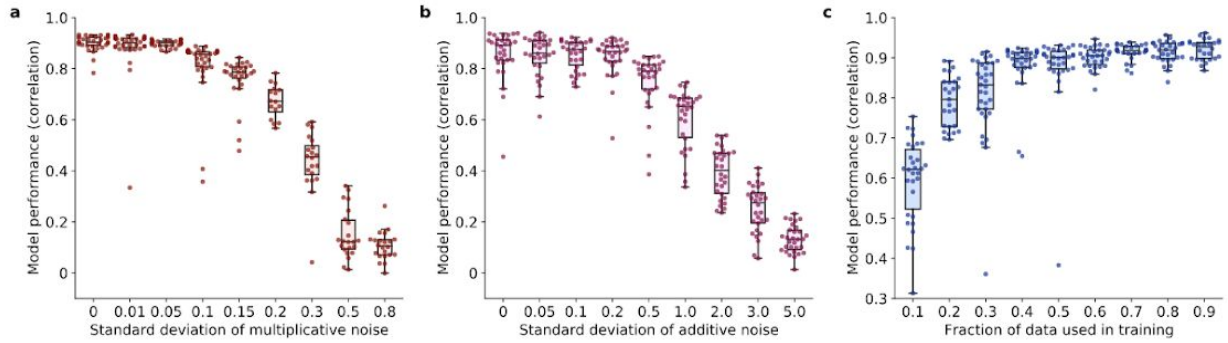


Figure 4. Model performance is stable against data noise and data reduction.

a-b. Correlation between predicted responses and experimental responses in the test set decreases as an increased level of multiplicative noise (**a**) or additive noise (**b**) is added to the training data (each dot represents one model). The CellBox models can tolerate up to σ_{mul} =0.05 multiplicative noise or σ_{add} =0.20 additive noise. **c.** Correlation between predicted responses and experimental responses in the test set increases with an increasing quantity of data used for model training. For the current dataset, the correlation plateaus when 40% of the original dataset is used.

Comparison of the network models with prior knowledge about pathways

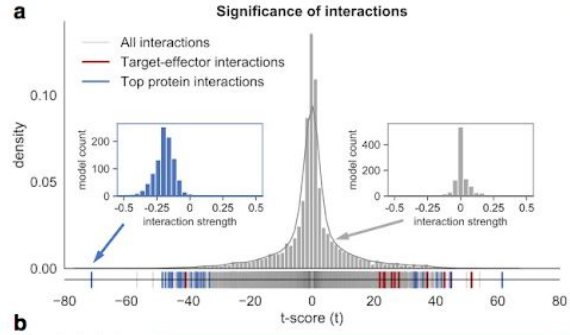
We used ordinary differential equations as the core framework of the current version of the CellBox mathematical model. Each parameter in the model represents the strength and direction of a biological interaction. In order to investigate whether the inferred interactions are consistent with current knowledge of biological pathways, we used the entire dataset as training data to generate 1,000 full models and examined the resulting *de novo* network edges learned from training. In order to measure both the strength and stability of edge inference, we used t-scores (see Methods) to assess the statistical

significance of each interaction, where a higher absolute t-score indicates higher interaction strength and lower variance across the models (Figure 5a). Using the primary targets of the drugs as the ground truth, we first examined the interactions between the drug-activity nodes and their downstream effectors. We found that all 12 drug-activity nodes had significant edge connections to their known primary downstream protein effectors with the interaction directions consistent with their expected effects (Figure 5b), suggesting the models were able to capture the literature-provided interactions between the drug target and their downstream effectors.

To further investigate to what extent the inferred networks represent known pathway interactions, we compared the interactions in the CellBox models to corresponding molecular interactions extracted from the Pathway Commons (PC) resource (Cerami *et al.*, 2011), which is an aggregation of ~20 curated publicly available pathway interaction databases. In this comparison between interactions inferred by CellBox models and prior knowledge, model interactions can, in principle, be found in PC either as one-step (A-B), two-step (A-X-B), or logical (>2 steps) interactions, where the logical interactions can be usefully predictive or perhaps erroneous. Direct interactions (A-B) in PC represent one component affecting the expression, phosphorylation, or state-of-change of the other component (B) (Figure. 5c, 5d, Supplementary Table S2). For example, models and prior knowledge agree for the phosphorylation of the AKT protein kinase (AKTpS473) that negatively affects the insulin receptor substrate 1 (IRS1) (Chandarlapaty *et al.*, 2011) and, for the activation of the mitogen-activated protein kinase (MAPKpT202) that affects p27 protein levels (Donovan, Milic and Slingerland, 2001; Osaki and Gama, 2013). Some model interactions can be found as indirect two-step interactions (A-X-B) in PC, meaning at least one path with one intermediate component(x) can be found between the two components (A and B). For example, retinoblastoma protein (Rb1) affects cyclinD through p21 (Carreira *et al.*, 2005; Lei, Liu and Ness, 2005); and mitogen-activated protein kinase kinase (MEK1) indirectly interacts in two steps with the transcription factor c-Myc by a phosphorylation

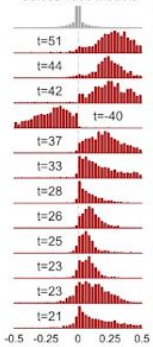
mechanism via ERK1/2 (Gupta and Davis, 1994; Butch and Guan, 1996; Sears, 2000; Aoki *et al.*, 2011). Additional evidence of functional links between proteins identified in the models includes protein-protein edges in the STRING database (von Mering *et al.*, 2005), such as links that represent expression (mRNA) correlation, and PubMed-derived links from text-mining (Maglott *et al.*, 2011).

In order to confirm that such agreement between the inferred networks and prior knowledge is not an artifact, we examined solution stability as well as compared the inferred networks to random networks. We conducted stability selection tests (Meinshausen and Bühlmann, 2010) on the interaction parameters, and the results indicate the solutions are reproducible and robust (Figure S9a). We compared the inferred networks to networks with randomly drawn interactions (see Methods). In the inferred networks, we observed a statistically significant higher number of top interactions consistent with the knowledge in Pathway Commons (PC), indicating that such agreement is not an artifact of the densely connected signaling networks in the pathway database (Figure S9b). The remaining model interactions are between components that are more than two pathway steps away or cannot be connected by any path in PC. These interactions can be interpreted as either logical interactions important for predictive purposes, or potential new physical interactions that are yet to be discovered in molecular experiments. Also, the Pathway Commons (PC) database aggregates interaction information from multiple biological systems, and, therefore, complete consistency is not expected for any particular system, e.g., the melanoma cell line used here. Since the CellBox network models are constructed in a completely data-driven way without any a priori intention to recapitulate known molecular interactions, the partial agreement between the model-inferred molecular interactions and the experimentally known ones from the literature is evidence of the validity of the modeling approach.



b

Strength distribution across 1000 models

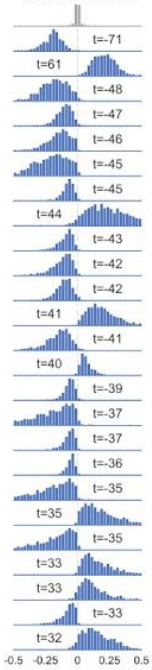


Drug target-effector interactions			
Drug target	Effector	Mean strength	Expected
amTOR	S6pS240	0.29	positive
aPI3K	AKTpS473	0.26	positive
aMEK	MAPKpT202	0.39	positive
aHDAC	p27	-0.23	negative
aBRAFm	MAPKpT202	0.27	positive
aCDK4	RbpS807	0.21	positive
aSTAT3	STAT3pY705	0.09	positive
aJAK	GSK3abpS21	0.11	positive
aMDM2	4EBP1pS65	0.08	positive
aSRC	GSK3abpS21	0.08	positive
aPKC	S6pS240	0.12	positive
aAKT	AKTpS473	0.18	positive

c

Top protein-protein interactions			
Upstream	Downstream	Mean strength	Interaction in PC
S6pS235	STAT5pY694	-0.20	-
S6pS235	PLK1	0.21	-
AKTpS473	AKT	-0.18	self
AKTpS473	IRS1	-0.10	direct
COX2	IGF1R-beta	-0.15	indirect
COX2	STAT3pY705	-0.21	direct
AKTpT308	STAT6pY641	-0.07	direct
4EBP1pS65	4EBP1pT37	0.25	self
4EBP1pS65	EGFRV	-0.10	-
4EBP1pT37	ACCpS79	-0.13	-
AKTpT308	BCL2	-0.11	direct
GSK3abpS21	TSC2pT1462	0.21	direct
S6pS240	Fibronectin	-0.15	-
GSK3abpS21	a-tubulin	0.07	indirect
S6pS240	CyclinE1	-0.07	-
MAPKpT202	p27	-0.20	direct
GSKpS9	EGFRV	-0.07	indirect
4EBP1pT37	ELK1pS383	-0.06	-
YBIpS102	p27	-0.22	-
YBIpS102	c-Myc	0.20	-
MAPKpT202	BIM	-0.18	direct
RbpS807	CyclinD1	0.18	indirect
GSKpS9	4EBP1pT37	0.13	direct
AKT	BIM	-0.08	direct
IRS1	S6pS240	0.15	indirect

Strength distribution across 1000 models



d

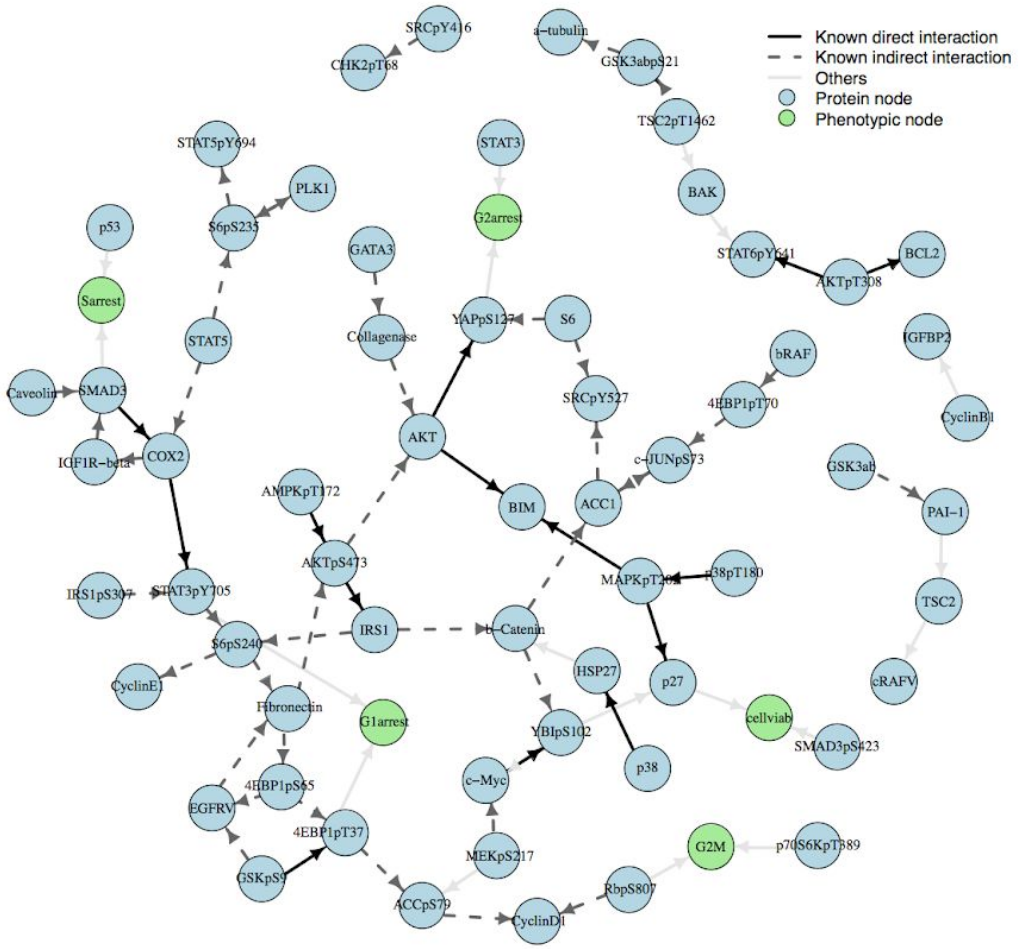


Figure 5. Comparison of the network models with prior knowledge about pathways

a. The t-score distribution of all interactions across 1,000 full models suggests that a small fraction of interaction strengths is significantly different from zero. Insets are two examples of interaction strength distributions across models. **b.** All 12 interactions between drug target (drug activity nodes) and their downstream effectors (red bars in **a**) are significant, and the interaction directions are consistent with the literature. **c.** Most of the top significant protein-protein interactions (blue bars in **a**) can be found as direct or indirect interactions in Pathway Commons (PC). The distributions of interaction strength across 1,000 models for each interaction in the two tables with corresponding colors are centered away from zero, in contrast to the background distributions of aggregated interactions across models (gray, all interactions with drug activity nodes in **a**, all protein-protein interactions in **c**). All other interactions with t-scores and PC information are included in Supplementary Table S2. **d.** A network visualization of the top interactions (top two interactions acting from each node and onto each node) highlights the level of agreement between model-inferred interactions and those in PC.

Predictions of unseen perturbations give candidates for drug combinations.

Our results so far indicate that the CellBox models can be efficiently trained on a relatively small set of experimental data to parametrize the differential equations that model the behavior of the entire system of nodes and interactions at a reasonable level of predictive accuracy. This model can then predict cell responses to a full range of single and combinatorial unseen perturbations, which would be laborious and costly to test exhaustively by experiments. In order to nominate effective drug combinations for a much reduced number of focused experiments, we used simulations of the 1,000 full models to quantitatively predict the dynamic cell responses to ~160,000 in silico

perturbations, including different dosages of single perturbations on each protein node as well as all pairwise combinations (see Methods). For each perturbation condition, we averaged the predictions across all models and ranked the perturbations by predicted phenotypic changes (Figure 6a).

Previous models on the same dataset, using the same differential equations but parametrized using belief propagation, had predicted that two drug pairs, MEKi+c-Myc and RAFi+c-Myc, would increase G1 cell cycle arrest and this prediction was confirmed by experiments (Korkut *et al.*, 2015). We found that the CellBox model predicts similar effects for these two drug pairs (Figure 6b). In order to identify additional therapeutic candidates, we examined the effects of all possible single and pairwise perturbations on cell cycle arrest (Figure 6b, 6c). The top-ranked candidates included dominant anti-proliferative inhibition (uniform colors in rows or columns) of proteins in the Wnt, MAPK, and ERK/MEK pathways, known to be cancer-related. Besides strong single candidates, synergistic drug pairs are of potential therapeutic interest (Figure 6b, 6c departure from uniform colors). Inhibitory perturbations predicted to have pro-proliferation effects, which are undesirable as such, can also lead to effective anti-proliferative candidates via indirectly activating perturbations (Figure 6c, top left corner). For example, protein nodes can, in principle, be activated by reducing upstream inhibition or degradation. As the CellBox model is completely data-driven, the *de novo* predictions represent system-specific predictions independent of prior knowledge.

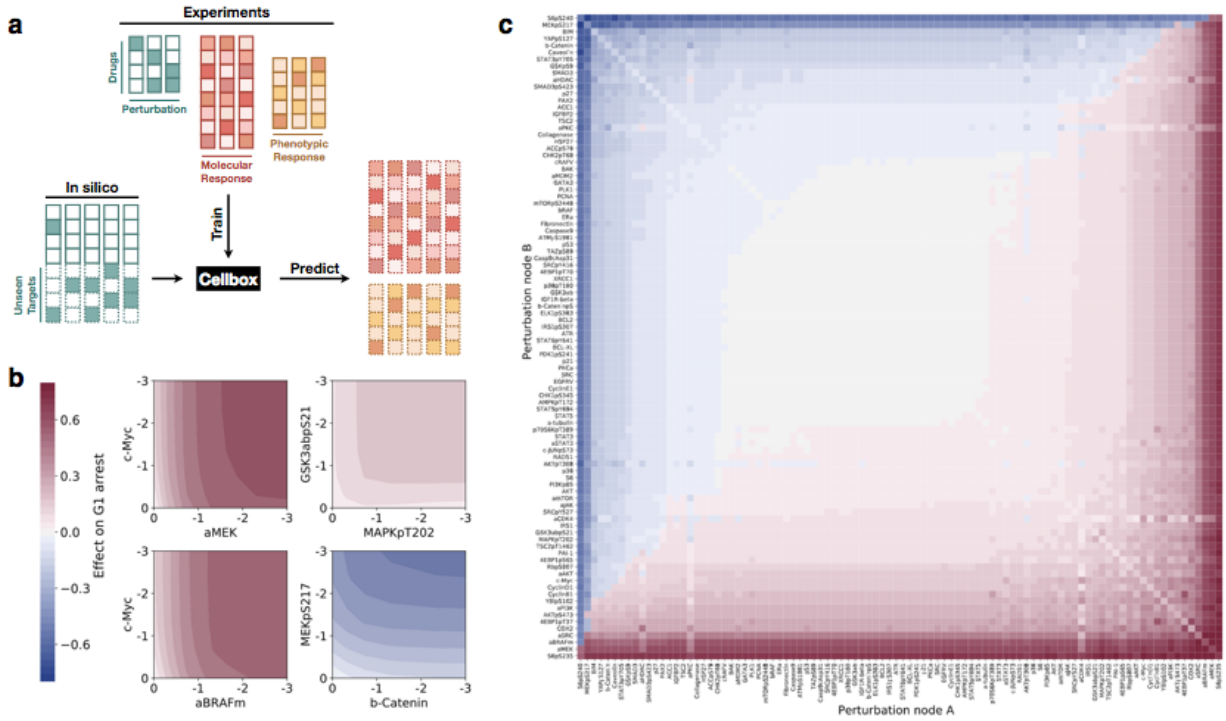


Figure 6. CellBox provides testable predictions of cell phenotype under synthetic perturbations.

a. For each (phospho)protein node in the network, CellBox was used to simulate all single and paired inhibitions and to predict the phenotypic changes . The phenotypic effects are the average prediction of 1,000 independent models trained on the full datasets. **b**. The anti-proliferation effects of two perturbation pairs whose effects on cell cycle arrest have been experimentally tested were closely examined (left two panels, c-Myc+MEKi, and c-Myc+RAFi), as well as two other in silico conditions (right two panels, GSK3p+MAPKp and MEKp+b-Catenin), by simulating with combinatorial perturbation strengths. **c**. The effects on cell cycle arrest of pairwise combinatorial perturbation of all (phospho)proteins in the network were simulated and used to nominate effective pharmaceutical candidates. These in silico inhibitory perturbations can result in anti-proliferation effects (red, bottom right) or pro-proliferation effects (blue, top left).

Discussion

Quantitative models that are predictive of dynamic cellular responses can be used to design combination therapies in cancer. To provide predictions with sufficient accuracy and potential mechanistic insight, we integrated machine learning methods with dynamic modeling: we applied an optimization algorithm used in deep learning to a biologically interpretable differential equation (ODE) system. Our model CellBox can be trained efficiently and independently of prior knowledge to predict molecular and phenotypic responses to unseen perturbations with high accuracy. Although trained on a relatively small set of experiments, the model is capable of simulating cell responses to numerous arbitrary combinatorial perturbations and dosages applied to nodes repeatedly measured under different perturbation conditions. Ranking of cellular responses to the in silico combinatorial perturbations by the desired phenotypic outcome, such as decreased proliferation, potentially leads to specific therapeutic hypotheses.

Interpretability of models that are to be used for practical decisions, such as the design of combination therapy, helps increase confidence and facilitates the design of focused validation experiments and is therefore as important as accuracy (Lipton, 2018). CellBox features interpretability in two aspects: transparency and traceability. *Transparency*: by using a well-defined mathematical model, CellBox is designed to be explicitly interpretable. In the current ODE model, each parameter represents a directed and quantitative interaction between cellular components or phenotypic quantities. *Traceability*: given a perturbation to the cellular system, the ODE simulation indicates how the effects of the perturbation propagate throughout the directed network in a time-dependent manner. The models can, therefore, provide mechanistic hypotheses of how the perturbations cause the observable cellular responses. Our model aims to meet the emerging demands of explainable artificial intelligence (XAI), which is essential for a

socially acceptable application of machine learning models in biomedicine (Thelisson, 2017; Gunning and Aha, 2019).

In principle, CellBox is generalizable to other types of systems and larger systems. Other types of models will presumably benefit from automatic differentiation (AD) combined with stochastic gradient descent that performs optimization directly for any given mathematical ansatz and, therefore, can avoid oversimplified approximations (Baydin *et al.*, 2018). The flexible AD framework allows the models to be easily adapted to various forms of cellular kinetics and dynamics. The ability to model larger systems depends both on the availability of larger datasets and scalable modeling methods. Larger datasets can be obtained by measuring diverse types of molecular data, for example, transcriptomic, epigenomic, and metabolomic changes (Brown *et al.*, 2014; Zaal and Berkers, 2018). A major opportunity for larger datasets may arise from recent cell barcoding techniques that significantly increase perturbation throughput relative to arrayed experiments (Adamson *et al.*, 2016; Dixit *et al.*, 2016) by measuring levels of transcripts by sequencing, levels of proteins detected by antibodies labeled by oligonucleotides or isotopes at the single cell level, or levels of both by multiplexing (Frei *et al.*, 2016; Wroblewska *et al.*, 2018; Mimitou *et al.*, 2019; Schraivogel *et al.*, 2020). A key advantage of single cell perturbation approaches for data-driven inference of dynamic networks is scale, as barcoded perturbation experiments can be pooled and individual cells identified based on sequence tags.

As CellBox is implemented in the Google TensorFlow framework, it can make use of various advanced machine learning techniques, such as dropout, mini-batching, and GPU boosting (Bengio, 2012; Liang and Liu, 2015), to improve training efficiency, which partially addresses the issue of scalability (Stapor *et al.*, 2019). As most state-of-art single-cell sequencing technologies still suffer from limited sequencing depth and inevitable noise, a new set of computational challenges arises when applying CellBox models to Perturb-seq like datasets. Future efforts are needed for resolving such issues

of sparsity and stochasticity, e.g., using robust techniques of dimensionality reduction (Lopez *et al.*, 2018; Eraslan *et al.*, 2019; Lotfollahi, Wolf and Theis, 2019).

A particular advantage of our modeling approach is its complete independence from prior knowledge. This is in contrast, e.g., to other frameworks for interpretable models (Fröhlich *et al.*, 2018) that incorporate prior knowledge by pre-defining the connections in the network model based on large-scale curation of molecular and biochemical pathways. Such models have been further developed and incorporated with proteome/phosphoproteome data (Schmiester *et al.*, 2020). While our current implementation of the model is completely data-driven, prior knowledge of cellular interactions could also be included by adding a penalty to the optimization function, which quantifies the disagreement between inferred and prior values for each parameter. However, considering the incompleteness of pathway knowledge bases and relatively high prediction accuracy of CellBox, we speculate that applying prior knowledge might give limited improvement in model performance. Besides, it would also be of interest to explore whether one can combine the efficiency advantage in Fröhlich et al. (Fröhlich *et al.*, 2018) via adjoint analysis (Fröhlich *et al.*, 2017) with the advantages of the current implementation of CellBox.

For broad translational applicability, a tantalizing but challenging prospect is to incorporate individual tumor background via genetic perturbations and to propose optimal, personalized combinations of targeted therapeutics. We envision the systems biology approach described here to be broadly applicable to other areas of biology, such as developmental biology or synthetic biology, provided that suitable perturbation-response data becomes available. Key future challenges are, therefore, the design of experiments for each biological context of interest and the further development of transferable and scalable machine learning methods.

Methods

Perturbation dataset overview

The CellBox models were trained using a perturbation-response dataset of the SK-Mel-133 melanoma cell line (Korkut *et al.*, 2015) (Figure S1). The cells were treated with 12 different single drugs, each at two different concentrations and 66 pairwise combinations of these drugs at IC40 concentrations. 24 hours after drug treatment, Reverse Phase Protein Arrays (RPPA) wasused to measure the level of 45 proteins and 37 phosphoproteins of interest. Cell cycle progression, including G1 arrest, G2 arrest, G2/M transition, and S arrest, was measured by flow cytometry. Cell viability was measured 72 hours after drug treatment by the resazurin assay. Our experimental data was normalized using median normalization, which is a standard approach of processing RPPA data (Tibes *et al.*, 2006). We used the $\log_2$ ratio of measurements in perturbed conditions over unperturbed conditions as the system variables x_i in the models. The dataset was initialized with 12 drug activity nodes representing the inhibition strengths of different drugs to their targets (Molinelli *et al.*, 2013). The resulting dataset has 89 perturbation conditions and 99 observed nodes (82 protein and phosphoproteins, 5 phenotypes, and 12 drug activity). A more detailed description of the experimental dataset is available in Korkut et al. (Korkut *et al.*, 2015)

Model configuration

The models were constructed using Python 3.6 and Google Tensorflow (Martín Abadi *et al.*, 2015) (version = 1.15.0). The molecular and phenotypic changes are linked in a unified biological network model using a system of ordinary differential equations

$$\frac{\partial x_i^{\mu}(t)}{\partial t} = \epsilon_i \phi\left(\sum_{j \neq i} w_{ij} x_j^{\mu}(t) - u_i^{\mu}(t)\right) - \alpha_i x_i^{\mu}(t) \tag{1}$$

where $x_i^{\mu}(t)$ represents the $\log_2$-normalized relative change of each (phospho)protein or phenotype levels relative to control levels under condition μ. $u_i^{\mu}(t)$ quantifies the strength of the perturbation on target (i). Here the drug effect is assumed to be constant and, therefore, $u(t) = u$ for $t > t_0$ is the perturbation strength determined using the endpoint level change of the primary target of the particular drug (Figure S1b). α_i characterizes the effect of decay, meaning the tendency of protein i to return to the original level before perturbation. The interaction parameters w_{ij} indicate interactions between network node j on network node i, assumed to be a constant property of the pair of molecules in this given cellular setting. We constrain the interaction parameters w_{ij} by disallowing three classes of interactions:

i) ingoing connections for drug nodes (drugs cannot be acted upon by any other node)

ii) outgoing connections for phenotypic nodes (phenotypes cannot act on any other nodes)

iii) self-interaction (nodes cannot act on themselves)

We use a sigmoid function $\phi(x) = tanh(x)$, to introduce a saturation effect of the interaction term so that it is bounded by the constant value of ε_i. Imposing bounds on the absolute value of the contribution of the interaction term to the derivatives is in part motivated by the fact that we are only modeling a small fraction of cellular components and by the experimental observation of system stability in these experiments. We also tested different alternatives of envelope forms, including a clipped linear function (hard tanh) $\phi(x) = max(-1, min(1, x))$, symmetric polynomial function $\phi(x) = \frac{\tilde{x}^n - 1}{\tilde{x}^n + 1}$, $\tilde{x} = max(0, x)$ (adapted from Hill's equation), and no envelope function $\phi(x) = x$ (linear) (Figure S5).

The biological network interactions were constructed *de novo* without any prior knowledge input, meaning the network was fully connected with interaction parameters.

The interaction parameters w_{ij} were randomly initialized following a normal distribution˜ $N(0.01, 1)$. The other two coefficients α_i and ε_i were initialized as 1.0.

Taken together, with this formulation of system dynamics, the effect of each new input perturbation is quantified by the response of downstream protein and phenotypic nodes, and the effect of the perturbation is simulated by propagating the input across the inferred interaction network according to the differential equations. After learning from perturbation-response data, the model can therefore make predictions for responses of perturbations on any experimentally probed node, including nodes not perturbed in the experimental dataset from which the models are derived. The ODE system was numerically solved using Heun's method (Süli and Mayers, 2003) (eqn. 2, time steps $N_t = 400$, supplementary Figure S2), which is an improved variant of Euler's method. Model performance was evaluated by disagreement between the experimental cell responses and the numerical steady state levels.

$$\tilde{x}_i(t+h) = x_i(t) + hf\big(t, x_i(t)\big)$$

$$x_i(t+h) = x_i(t) + \frac{h}{2}\left[f\big(t, x_i(t)\big) + f\big(t_{i+1}, \tilde{x}_i(t+h)\big)\right] \quad (2)$$

Where h is the step size, $f\big(t, x(t)\big) = x'(t)$, $y(t_0) = \log(1) = 0$.

The loss function $L(w)$ is defined as a weighted sum of prediction error and complexity penalty in order to avoid overfitting. Here a mean squared error (MSE) and an L1-loss regularization term are used, as defined in eqn. 3. We have tested different regularizations, including L1, L2, and both combined (elastic net) with various strengths (regularization strengths for L1, L2 term λ_1, λ_2 = 0, 0.01, 0.0001, pairwise combination), and found no significant difference in model performance (Figure S2b). The interaction

parameters were optimized end-to-end using the Adam optimizer (Kingma and Ba, 2014), with the objective of minimizing the loss function.

$$L(\mathbf{w}) = \sum_{\mu} \sum_{i} ||\hat{x}_i^{\mu}(\mathbf{w}, t) - x_i^{\mu *}||_2^2 + \lambda ||\mathbf{w}||_1 \qquad (3)$$

$\hat{x}_i^{\mu}(w, t)$ is calculated as the converged value of the numerical simulation of the ODE with the interaction parameters w and defined simulation timestep t. $x_i^{\mu *}$ indicates experimental measurement which is used as a gold standard in training. The dataset was divided into training, validation, and test sets, in order to optimize parameters, provide an indication for stopping training, and test model performance, respectively. Optimization was conducted with an initial learning rate for the Adam optimizer (lr=0.1) and regularization strength (λ=0.01). It has been shown that a gradually decreasing learning rate is helpful for model convergence (Bengio, 2012). The model training was stopped when the loss function of the validation set does not further decrease for a continuous of 20 iterations (stopping patience).

The model was trained with mini-batching: a random 80% portion of the training set was used to optimize parameters for each iteration, and a fixed batch size of 4 was used in each epoch. Models that failed to converge (MSE for training set > 0.05) were excluded as unsuitable. For a larger dataset, we recommend using a fixed batch size of 8 or 16, as documented in the latest version of the software on GitHub.

Model training with random data partitions

For initial model training and analysis of model performance, the cell line perturbation-response dataset was randomly partitioned into training, validation, and test set in the proportion of 56% (n=50 conditions), 14% (n=12 conditions), and 30% (n=27 conditions). 1,500 models were generated on 1,500 independently random-partitioned datasets.

The models were examined and categorized into non-oscillating and oscillating solutions based on time derivatives at the final time step of the ODE simulation. The non-oscillating solutions are defined as those with the average absolute value of time derivatives of all nodes and conditions in the training set smaller than δ, i.e. $\frac{1}{m}\sum_{i=1}^{m}\left|\frac{\partial x_i^{\mu}(t)}{\partial t}\right| < \delta\,;\delta = 1e-03$. In each category, twenty models were randomly selected, and each re-trained with the original data partitioning but forty different random seeds, covering all the random processes in training, including parameter initialization and mini-batching sampling (Figure S4). Oscillating solutions comprise about 30 percent of all models. In the following analysis, models that converged to oscillating solutions were excluded.

Under the training scheme using random data partition, we examined the sensitivity to data scaling and to the choice of ODE solver, by applying the following modifications to the data or the model and evaluating the changes of model performance. i) In addition to the $\log_2$ scale of the ratio of measurements in perturbed conditions over unperturbed conditions, we applied other scalings to the raw data to generate model input, including non-log (linear) scale (Figure S3c). ii) In addition to Heun's ODE solver, we tested other numerical ODE solvers, including Euler methods, Midpoint methods, and Runge-Kutta methods (Figure S4e-g).

Single-to-combo and leave-one-drug-out cross-validations

In the single-to-combo task, all single-drug conditions were allocated to the training set (n=23 conditions), and the combination perturbation conditions were randomly distributed among the validation and test set (n=53 conditions) in a 20/80 ratio. To evaluate model performance by cross-validation for each drug, the data was partitioned into training (n=78 conditions) and test (n=11 conditions) sets, where each test set contains all the drug combination conditions with the particular drug. 20% of training conditions (n=15) are used as a validation set. The predictions on the test set were

averaged over 100 models. In the more stringent leave-one-drug-out task where all perturbation conditions involving one drug were withheld as the test set, the evaluation of model performance was conducted by applying perturbation of the same strength directly on the downstream effector node of each withheld drug activity node (Figure S1b).

The Belief Propagation (BP) models for both single-to-combo and cross-validation prediction were performed as in our earlier publication (https://github.com/korkutlab/pertbio). The predictions on the test set were averaged over 100 models. The deep neural network model (NN) network had 2 densely connected hidden layers (hidden layer H1: 20 neurons, H2: 100 neurons), connecting the parameterized perturbation tensor with the cell response tensor. We used the *tanh* activation function for the hidden layers and the leaky rectified linear unit function (ReLU) for the output layer. The NN models were constructed in the Tensorflow framework in Python and optimized using the same optimization methods (Adam optimizer). The co-expression static model (Co-exp) was constructed in the Python environment using the sklearn (version = 0.21.3, https://scikit-learn.org/stable/) *MultiTaskElasticNetCV* module, which is a linear regression model with L1 and L2 regularizations whose strengths are automatically determined using built-in cross-validation. The model was trained to use the changes in levels of each pair of protein nodes (input) to predict levels for the rest of the proteins and phenotypes (output). To compare the four different predictive models, CellBox, BP, NN, and Co-exp, the paired t-test on two related samples was used to analyze the difference between model performance (test set correlation) and to assign p-values.

Sensitivity analysis with noise and reduced training set size

We conducted a sensitivity analysis of our model to evaluate the robustness of its prediction in response to noise. To model experimental noise, we first applied varying levels of multiplicative Gaussian noise to the input molecular and phenotypic data (eqn.

4). The assumption is that the uncertainty and noise in experimental measurements arise around the true values with a scaling factor depending on the experimental approach.

$$x_i(t) = x_i^*(t) * N\,(1, \sigma_{mul}) \quad (4)$$

The scaling factor for each node and each condition was independently drawn from a Gaussian distribution $N\,(1, \sigma_{mul})$, with a mean of 1 and a standard deviation of σ_{mul}. $x_i^*(t)$ represents the experimental measurements, and $x_i(t)$ represents the values with noise added. For each noise level, we evaluated 15 different training/validation/test partitioning and each with 5 independent random noise patterns. Model training was performed on noisy training and validation sets, while the model performance evaluation was performed on the original, noise-free test data. For each noise level, the percentage of successful models, defined as those that converged in terms of both MSE and oscillation filters, was recorded.

To further evaluate model performance from a technical perspective, we applied additive Gaussian noise to the input data (eqn. 5) and evaluated model performance as above.

$$x_i(t) = x_i^*(t) + N\,(0, \sigma_{add}) \quad (5)$$

We examined model sensitivity to training and validation set size. We reduced the combined size of the training and validation set, from 90% to 10% in steps of 10%, while keeping their relative size constant, 4:1. The remaining data were allocated to the test set. For each training set size, the percentage of successful models, defined as those that converged in terms of both MSE and oscillation filters, was reported.

Biological interpretation of the network model

The entire dataset was used to generate 1,000 full successful models, each with an independent data partitioning of training (n=71 conditions) and validation (n=18 conditions). For each interaction (w_{ij}) between two nodes, a t-score ($\frac{\bar{w}_{ij}}{s\sqrt{m}}$) (which is

effectively an one-sample t-test with the null hypothesis that the population mean is zero) was calculated as an indication of the confidence level of obtaining a value different from zero, where $\bar{w}_{ij}$ is the average interaction strength across models, s is the standard deviation, m is the number of models (m = 1,000) (Supplementary Table S2).

In order to compare the model inferred interactions to those present in prior-knowledge pathway databases, all the proteins and phosphoproteins nodes were identified by their corresponding gene names (Supplementary Table S1). The interactions were compared against the Pathway Commons (PC) database (current version at https://www.pathwaycommons.org/archives/PC2/v11) using the paxtoolsr (Luna *et al.*, 2016) software. The database was filtered down to direct interactions, which include "controls expression of", "controls phosphorylation of", "controls state change of", "controls production of", "controls transport of", and "controls transport of chemical". All the direct interactions were converted to a directed PC graph using igraph (version = 1.2.4.1, https://igraph.org/r/). Distance was calculated as the length of the shortest path (s) between two components in the PC graph (Supplementary Table S2, column: distance). The number of the shortest paths (npath) between the two components was also reported. Additional detailed information on protein links in Homo sapiens was obtained from the STRING database, including combined score (string) and individual-channel scores (co-expression, experimental, database, text-mining) (https://stringdb-static.org/download/protein.links.detailed.v11.0/9606.protein.links.detailed.v11.0.txt.gz) (von Mering *et al.*, 2005). The co-citation score was calculated as the number of papers mentioning both components in a customized paper collection aggregating all PubMed papers that refer to at least two and no more than five gene names (Maglott *et al.*, 2011).

To quantitatively evaluate whether the inferred networks recapitulate more known interactions in the database than the random networks, the same number of network

models (m = 1,000) were generated with interaction parameters randomly drawn from the pool of all (phospho)protein- (phospho)protein interaction parameters in the CellBox-inferred network models. For each of the inferred or random network models, the number of interactions in Pathway Commons (PC) out of the top one-hundred interactions (ranked by absolute interaction strengths) was calculated and the t-test was performed to compare the difference in means between the two groups. To provide an additional comparison with a well-known alternative method for network inference, the partial information decomposition and context (PIDC) algorithm was implemented in the Python environment based on the detailed mathematical description of the algorithm in Chan et al (Chan, Stumpf and Babtie, 2017). The same number of PIDC networks (m = 1,000, same as for CellBox) were generated based on 1,000 independent random partitions of the entire dataset (80% of the conditions were used to infer PIDC networks). The t-test was performed to compare the difference in means between the number of interactions inferred by CellBox network models and PIDC networks consistent with the PC database (Figure S9).

Model predictions for a large number of unseen perturbations

We used the models trained with the full (non-partitioned) dataset to simulate responses of novel, experimentally unobserved, in silico perturbation conditions. These conditions included different doses of single perturbations (different levels of perturbation strength $u \in [0, 3]$ on all individual (phospho)protein and drug activity nodes within the network and as well as all pairwise combinations ($n_{all} = 94 \times 6 + C_{94}^{2} = 157,920$ conditions). The cell responses were dynamically simulated with u as the input perturbation with the same number of steps as in training (N_t = 400). For each perturbation condition, predictions for cell responses were averaged across 1,000 different models. Perturbations were nominated as therapeutic candidates by ranking the predicted magnitude of the phenotypic change in terms of cell cycle arrest.

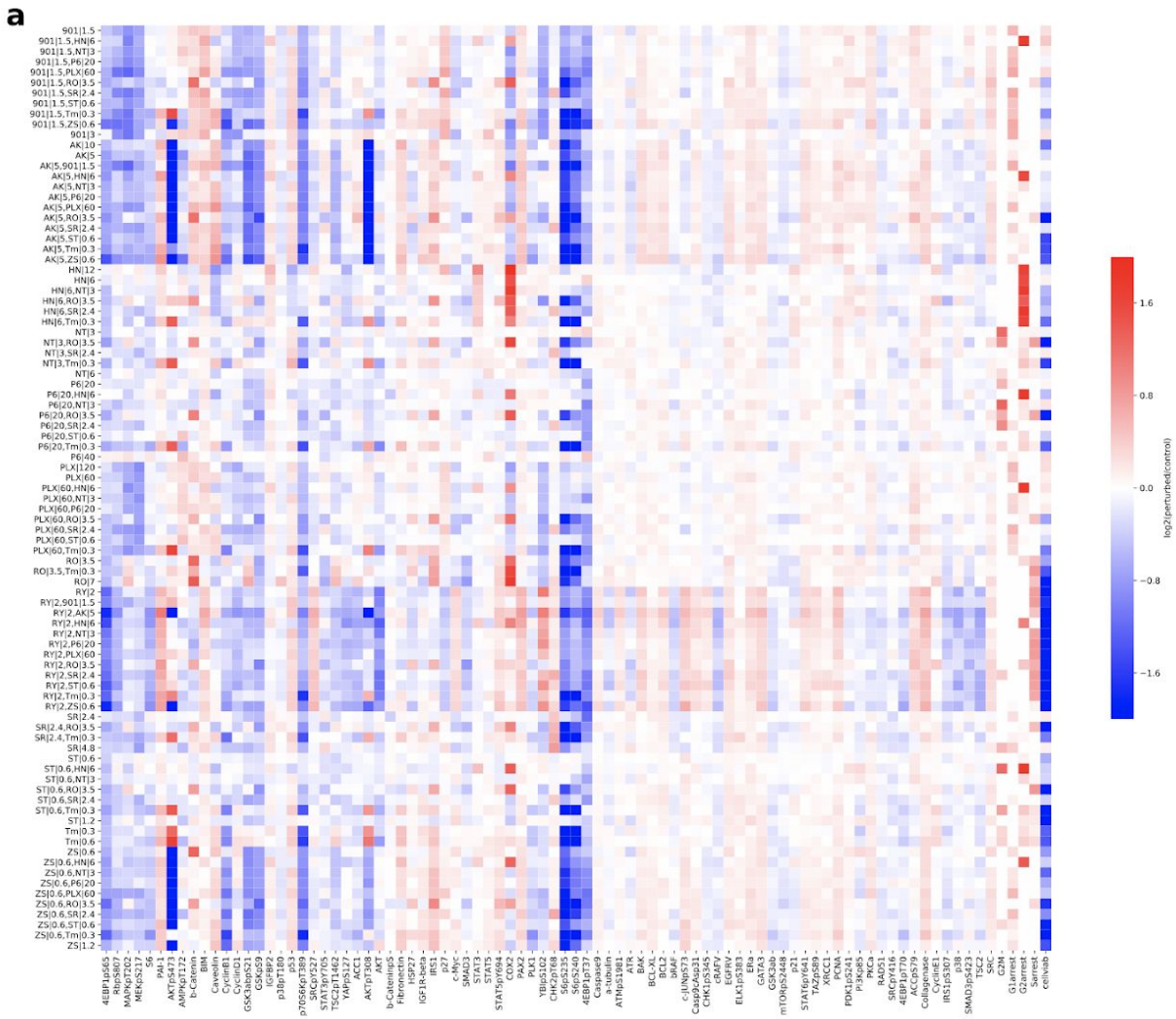
Code and data availability

CellBox code and data are available at https://github.com/dfci/CellBox as release v0.2. We provide a shareable, online, interactive environment using Binder with all necessary dependencies pre-installed. The Quick Start section provides instructions to quickly try an example script that runs a shortened training process.

Acknowledgements

We thank Alexandra Franz, Frank Poelwijk, Laura Kleiman, Nicholas Gauthier, Haozhe Shan, William Yuan, Han Altae-Tran and members of the Sander and Marks labs for constructive discussions. Funding support from DFCI, NHGRI (U41HG006623) and NIGMS (P41GM103504).

Supplementary materials

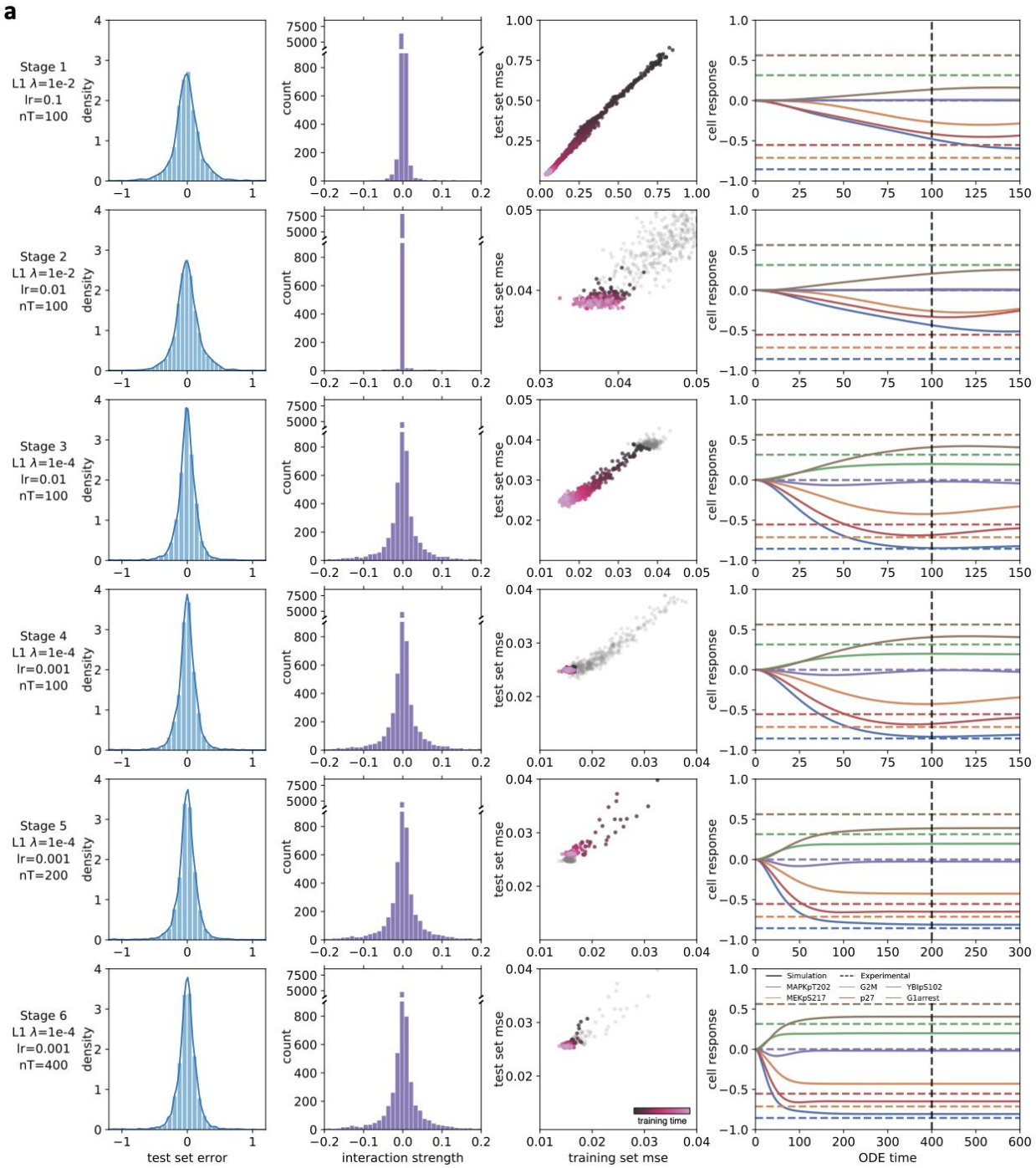


b

Drug used in perturbation experiments					
Abbreviation	**Drug name**	**Target**	**Downstream effector**	**Dose 1 (nM)**	**Dose 2 (nM)**
AK	AKTi 1/2	AKT	AKTpS473	5.0	10.0
Tm	Temsirolimus	mTOR	S6pS240	0.3	0.6
ZS	ZSTK474	PI3K	AKTpS473	600	1,200
901	PD0325901	MEK1/2	MAPK1pT202	1.5	3.0
ST	Stattic	STAT3	STAT3pY705	600	1,200
HN	HNHA	HDAC	P27/Kip1	6,000	12,000
PLX	PLX4720	$BRAF^{V600E}$	MAPK1pT202	60	120
RO	RO-31-7549	PKC	S6pS240	3,500	7,000
RY	RY	CDK4	RbpS807	2,000	NA
P6	P6	JAK	GSK3pS21	20	40
SR	SR	SRC	GSK3pS21	2,400	4,800
NT	Nutlin	MDM2	4EBP1pS65	3,000	6,000

Figure S1. An overview of the perturbation dataset.

a. The perturbation dataset ($\log_2$ transformation of the ratio of measurements in perturbed conditions over unperturbed condition) was visualized in a heatmap with rows of perturbation conditions (format: drug abbreviation with concentration, single or pair of drugs) and columns of (phospho)proteins or phenotypic measurements. **b.** 12 drugs were used in the perturbation experiments, and the table contains for each drug their abbreviations, targets, downstream effectors used in the inference of drug activity nodes, and the two doses used in single-drug perturbation conditions. In combinatorial perturbation conditions, the lower dose of each drug was combined.


a
Stage 1
L1 λ=1e-2
lr=0.1
nT=100
Stage 2
L1 λ=1e-2
lr=0.01
nT=100
Stage 3
L1 λ=1e-4
lr=0.01
nT=100
Stage 4
L1 λ=1e-4
lr=0.001
nT=100
Stage 5
L1 λ=1e-4
lr=0.001
nT=200
Stage 6
L1 λ=1e-4
lr=0.001
nT=400
density
count
test set mse
cell response
Simulation
Experimental
MAPKpT202
G2M
YBlpS102
MEKpS217
p27
G1arrest
training time
test set error
interaction strength
training set mse
ODE time

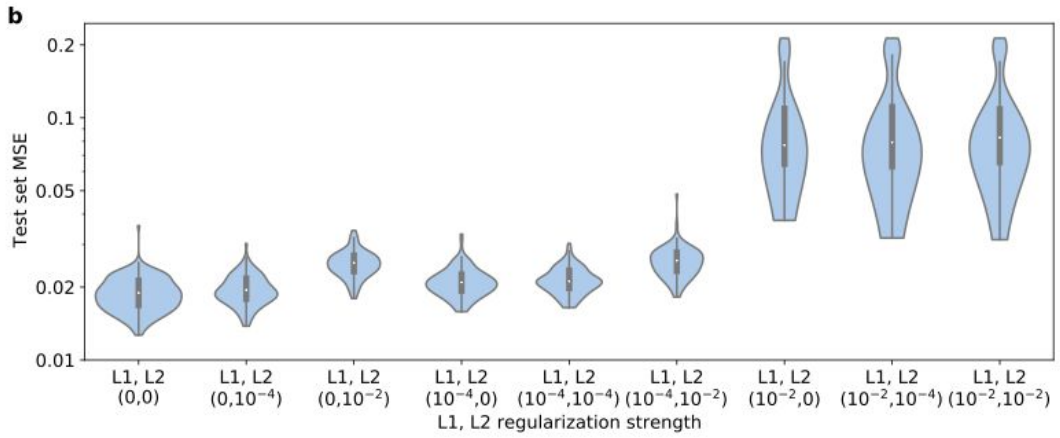
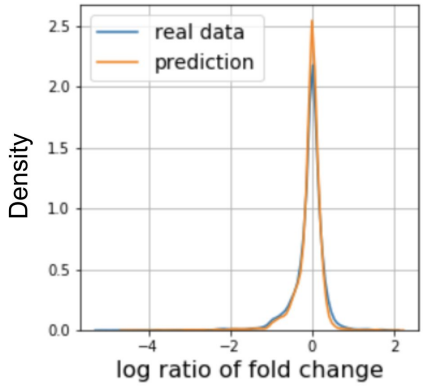


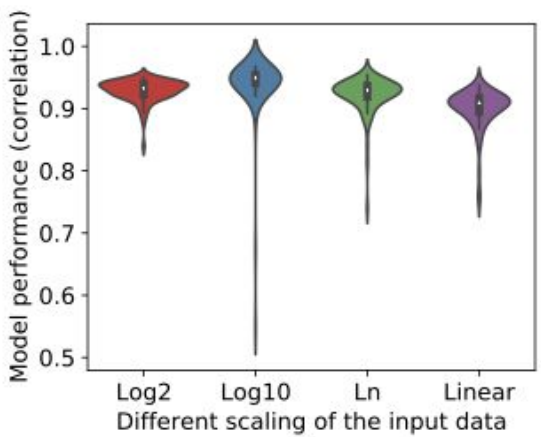
Figure S2. Multi-step fine-tuning of hyperparameters facilitates model training. During the training of each model, the training loss decreased along training time together with the test loss (Stage 1). The model parameters started to fluctuate around a local optimum after N = 2,000 training iterations. It has been shown that a decreasing learning rate is helpful for model convergence (Bengio, 2012). Decreasing the learning rate to 0.1x allowed the model to escape local minima and continue learning (Stage 2). The loss function stopped decreasing again after (up to) N = 4,000 iterations when the magnitude of the MSE loss was comparable to that of the regularization loss. To further improve training and decrease MSE loss mainly, we decreased the regularization strength by loosening the L1 constraints on the parameters (Stage 3). MSE decreased further while the numerical range of the parameters (interaction strengths) started to increase. Continuous decreasing of the learning rate did not further change the loss (Stage 4). ODE simulation of the model indicated that a steady state had not been fully reached. Therefore, the ODE simulation time was doubled twice (Stage 5 and Stage 6) while the learning rate and regularization were kept the same as Stage 4. The training and testing loss, together with the ODE trajectory, indicated that, at the final stage, the models had converged, optimization made no further improvements, and the ODE simulation has reached a steady state. We then stopped the training and examined the results closely on the test dataset. **a.** Models were trained in six individual stages with varying learning rate, regularization strength, and ODE simulation time. As the training proceeds, 1. the distribution of differences between predicted and experimental values

in the test set narrowed around zero; 2. the distribution of interaction strengths widened as the L1 regularization was weakened; 3. both the training and test loss decreased; 4. ODE simulation reached a steady state as simulation time increased. **b.** Model performance was evaluated on different regularization approaches, including L1, L2, or combined (elastic net) with different regularization strengths (regularization strengths for L1, L2 term λ_1, λ_2 =0, 1e-2, 1e-4, pairwise combination). For a mild level of regularization ($\lambda_1 =$ 1e-4 and $\lambda_2 =$ 1e-4), we observed no significant difference between L1, L2, and elastic net. A stronger L1 regularization compromises the accuracy of model predictions.

a


real data
prediction
Density
log ratio of fold change

c


Model performance (correlation)
Log2
Log10
Ln
Linear
Different scaling of the input data

b

Figure S3. Correlations between predicted and experimental data were consistently high across different perturbation conditions.

a. Model prediction and experimental data had a similar range and distribution without skewing and extreme predictions. **b**. In addition to overall performance, the model predictions for each perturbation condition were examined. The prediction of cell response under each condition reached similarly high correlation (median Pearson's correlation 0.95) with experimental data. Meshes indicate the range of real data. Models generally performed better for conditions with larger data range. **c.** After median normalization of the RPPA data, the $\log_2$ ratio of measurements in perturbed conditions over unperturbed conditions was calculated and used as the input for the model. To evaluate whether the model performance depends on our choice of logarithmic scaling, we tested the model robustness against different scaling, including $\log_{10}$, natural log (ln), and non-log (linear) scaling. The model performance remains similar for different logarithmic bases (Pearson's correlation coefficient for test set $\rho>0.93$). Without logarithmic transformation, the model training still converges and results in a slightly reduced predictivity ($\rho=0.90$). The performance of CellBox is robust against different scaling methods.

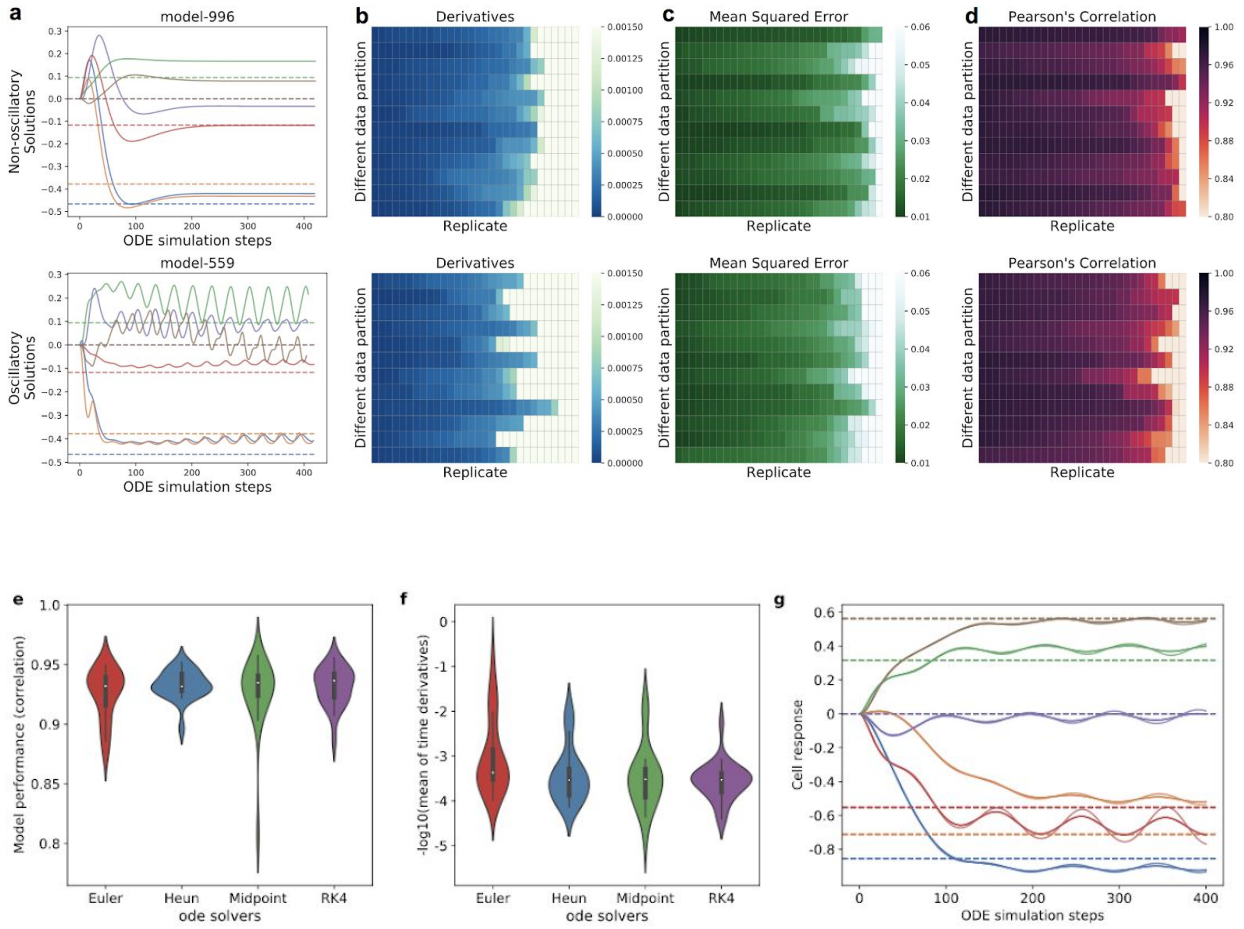


Figure S4. Oscillatory models from stochastic training are independent of data partitioning.

a. Models were examined and categorized into non-oscillatory and oscillatory solutions based on the ODE simulation trajectories. **b-d.** For each data partitioning of training and test set (rows) in the two categories, different seeds for random processes in model training (columns). The models were examined for their performance in terms of average derivatives of each variable at the end of the ODE simulation (**b**), average mean squared error in the training set (**c**), and Pearson's correlation between prediction and experimental data (**d**). Therefore, the solution oscillation and the model convergence are independent of the data partition. **e.** The model performance in terms of Pearson's correlation between prediction and experimental data remains similar when different ODE solvers, including Euler, Heun, Midpoint and Runge-Kutta (RK4),

were used in CellBox models. **f.** Oscillatory solutions (mean of time derivatives threshold $\delta = 1e-03$) exist when all different ODE solvers are used. **g.** For a given solution, all solvers are consistent in terms of whether the solution is oscillatory or not but might differ slightly in terms of the oscillation amplitude.

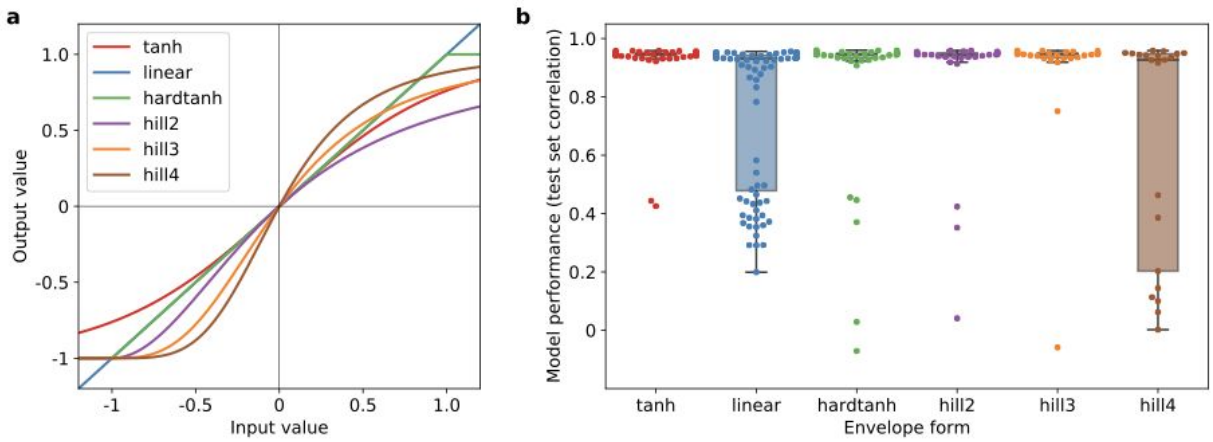


Figure S5. Model performance with different envelope forms

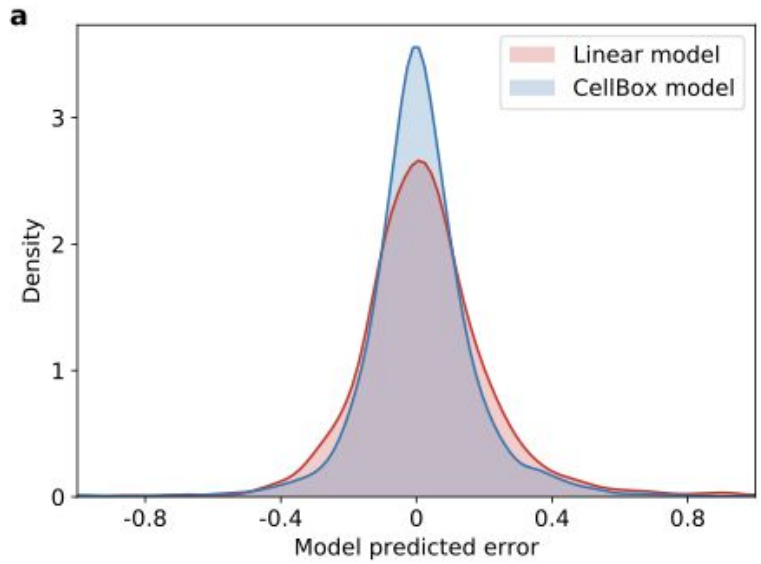
a. We tested different alternatives of envelope forms, including a hyperbolic *tanh* (used as current default in CellBox), a clipped linear function (hard *tanh*), symmetric sigmoid function (hill of various degrees), and no envelope function (linear). All the other envelope forms model the saturation effect except for the linear function. **b**. We observed comparable prediction accuracy between *tanh*, clipped linear, and low order of sigmoid functions, while the training with a linear envelope and high order of sigmoid functions was much less stable, and prediction performance was less accurate on average .

		Belief Propagation	Co-expression	Neural Network	CellBox
Performance	Training time* (CPU only)	30-40 mins	**5 mins**	**5 mins**	10-20 mins
	Training time* (GPU)	N/A	N/A	**<1 min**	4-8 mins
	Correlation (Leave-one-drug-out)	0.80	0.92	0.92	**0.94**
	Correlation (Single-to-combo)	0.69	0.84	0.88	**0.92**
Interpretability	Framework	**Biological mechanism**	Correlation	Correlation	**Biological mechanism**
	Dynamics	**Traceable**	-	-	**Traceable**
Prediction	Cooperative effects (non-linear)	**Yes**	-	**Yes**	**Yes**
	Experimentally untested targets	**Yes**	-	-	**Yes**

* Model training is timed for network size m = 99, using 4 vCPUs and 2GB RAM, with an optional Nvidia P100 GPU.

Figure S6. Comparison of various features between CellBox and other models.

In addition to the comparison of model performance in several training schemes between different models (Figure 3), we provided a more detailed comparison between models in the perspectives of training time and model capabilities and labeled the best features out of all models (bold). Generally, we believe the faster the training is with a similar number of inferred parameters, and the more biological mechanisms the model can provide, the more generalizable and useful the model is.



b

Examples of non-additive effects in drug combinations							
Combination	**Effector**	**Drug A effect**	**Drug B effect**	**Effect if additive**	**Measured effect**	**CellBox prediction**	**Effect**
AKTi+PI3Ki	b-Catenin	0.25	1.28	1.51	0.42	0.51	antagonistic
PI3Ki+PKCi	b-Catenin	1.28	1.26	2.52	0.76	1.03	antagonistic
PI3Ki+SRCi	S6pS235	-1.18	-0.22	-1.15	-2.26	-2.25	synergistic
JAKi+PKCi	Cell viability	-0.07	0.10	-0.33	-1.88	-2.07	synergistic
SRCi+PKCi	Cell viability	-0.05	0.01	-0.24	-2.08	-2.36	synergistic

Figure S7. CellBox predicts the non-additive effects of drug combinations.

One of the most important applications of CellBox is to raise candidates of synergistic drug combinations. In order to test model prediction for synergy, we examined the model predictions in the single-to-combo training scheme where only the single-drug perturbation conditions were used to predict cell response to drug combinations. We compared those predictions with additive predictions (linear model) and observed CellBox models could predict both synergistic and antagonistic effects. **a**. A systematic comparison of the prediction from CellBox and linear additive models. The CellBox models predict with higher accuracy (smaller prediction errors). The error is defined as

the difference between model predictions and experimental responses for each data point. **b**. Examples of non-additive effects were chosen from the top ranking list of difference between prediction error from linear model and that from CellBox models, which are effectively the proteins or phenotypes in drug combinations that CellBox predicts accurately while the linear model does not.

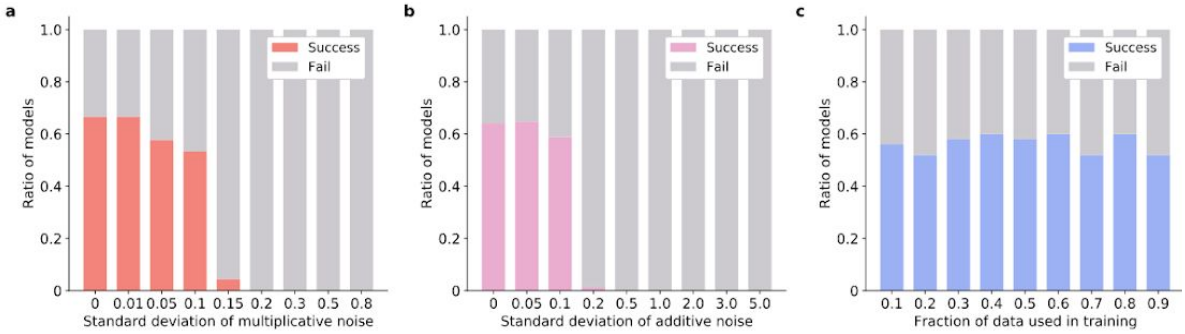


Figure S8. Model convergence against noise and reduced training set

a-b. The percentage of models that successfully converged, defined by MSE of training set below a threshold of 0.05, decreases as an increased level of multiplicative Gaussian noise (**a**) or additive Gaussian noise (**b**) was added into training data. Such a decrease in model accuracy is plausible: as the signal is overwhelmed by the increasing noise level, model performance and stability should decrease in terms of MSE. **c.** The percentage of successful models stayed the same as more data was used for model training.

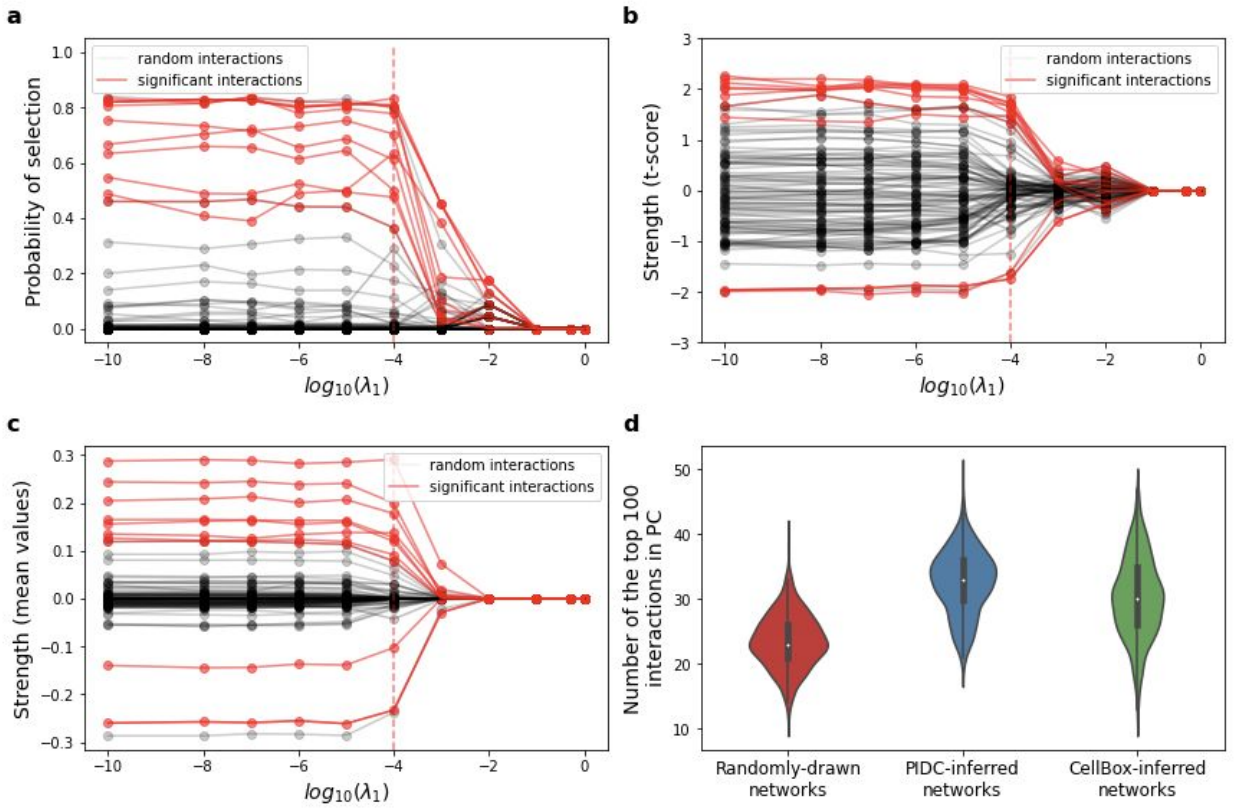


Figure S9. De novo inferred interactions are both robust and significantly consistent with literature derived networks.

a-c. We used stability selection approaches (Meinshausen and Bühlmann, 2010) to examine network inference stability. We trained CellBox models with different L1 regularization strengths λ_1 and plotted the stability paths, including the probability of selection (**a**), t-score of the interactions over the models (**b**), and mean values of the interactions over the models (**c**), to compare the top 100 interactions (10 out of these are highlighted in red lines) identified by CellBox to 100 randomly chosen interactions from the pool of all possible (phospho)protein- (phospho)protein interactions (grey lines). The CellBox models result in significantly stable parameter inference, compared to random permutations (m>140 models for each λ_1). **d.** To quantitatively evaluate our models in the context of prior knowledge, we compare CellBox models to random models: the same number of network models (m = 1,000) were generated with interaction parameters randomly drawn from the pool of all (phospho)protein-

(phospho)protein interaction parameters in the CellBox-inferred network models. For each of the inferred or random network models, the interactions existing in PC were identified out of the top one-hundred interactions (ranked by absolute interaction strengths). Our results find a significant difference in the number of interaction edges consistent with prior knowledge from Pathway Commons between the CellBox-inferred network and a network with random interactions (t-test, p=3e-149, $N_{\text{CellBox-inferred interactions in PC}} - N_{\text{random interactions in PC}} = 7$). We also compared our methods with other network inference methods (Chan, Stumpf and Babtie, 2017). Partial information decomposition and context (PIDC) recovers more prior knowledge interactions in PC compared to CellBox (by t-test, p-value=8.0e-21, $N_{\text{PIDC-inferred interactions in PC}} - N_{\text{CellBox-inferred interactions in PC}} = 2$). Nevertheless, we argue that the slight advantage of PIDC compared to CellBox in this measure is more than compensated for by the crucial ability of the dynamically executable CellBox model to predict cell response to unseen perturbations, which is the primary objective of our approach.

Supplementary Tables

Table S1. Annotation of nodes in the network.

Table S2. Model-inferred molecular interactions and their pathway database information

References

Adamson, B. *et al.* (2016) 'A Multiplexed Single-Cell CRISPR Screening Platform Enables Systematic Dissection of the Unfolded Protein Response', *Cell*, 167 (7), pp. 1867–1882.e21.

Aldridge, B. B. *et al.* (2009) 'Fuzzy logic analysis of kinase pathway crosstalk in TNF/EGF/insulin-induced signaling', *PLoS computational biology*, 5 (4), p. e1000340.

Aoki, K. *et al.* (2011) 'Processive phosphorylation of ERK MAP kinase in mammalian cells', *Proceedings of the National Academy of Sciences of the United States of America*, 108 (31), pp. 12675–12680.

Azmi, A. S. *et al.* (2010) 'Proof of concept: network and systems biology approaches aid in the discovery of potent anticancer drug combinations', *Molecular cancer therapeutics*, 9 (12), pp. 3137–3144.

Babur, O. *et al.* (2010) 'Discovering modulators of gene expression', *Nucleic acids research*, 38 (17), pp. 5648–5656.

Baydin, A. G. *et al.* (2018) 'Automatic Differentiation in Machine Learning: a Survey', *Journal of machine learning research: JMLR*, 18 (153), pp. 1–43.

Bengio, Y. (2012) 'Practical Recommendations for Gradient-Based Training of Deep Architectures', *Lecture Notes in Computer Science*, pp. 437–478. doi: 10.1007/978-3-642-35289-8_26.

Brown, R. *et al.* (2014) 'Poised epigenetic states and acquired drug resistance in cancer', *Nature reviews. Cancer*, 14 (11), pp. 747–753.

Bruggeman, F. J. *et al.* (2002) 'Modular response analysis of cellular regulatory networks', *Journal of theoretical biology*, 218 (4), pp. 507–520.

Butch, E. R. and Guan, K. L. (1996) 'Characterization of ERK1 activation site mutants and the effect on recognition by MEK1 and MEK2', *The Journal of biological chemistry*, 271 (8), pp. 4230–4235.

Carreira, S. *et al.* (2005) 'Mitf cooperates with Rb1 and activates p21Cip1 expression to regulate cell cycle progression', *Nature*, 433 (7027), pp. 764–769.

Carter, S. L. *et al.* (2004) 'Gene co-expression network topology provides a framework for molecular characterization of cellular state', *Bioinformatics* , 20 (14), pp. 2242–2250.

Cerami, E. G. *et al.* (2011) 'Pathway Commons, a web resource for biological pathway

data', *Nucleic acids research*, 39 (Database issue), pp. D685–90.

Chandarlapaty, S. *et al.* (2011) 'AKT inhibition relieves feedback suppression of receptor tyrosine kinase expression and activity', *Cancer cell*, 19 (1), pp. 58–71.

Chan, T. E., Stumpf, M. P. H. and Babtie, A. C. (2017) 'Gene Regulatory Network Inference from Single-Cell Data Using Multivariate Information Measures', *Cell systems*, 5 (3), pp. 251–267.e3.

Cheng, F., Kovács, I. A. and Barabási, A.-L. (2019) 'Network-based prediction of drug combinations', *Nature communications*, 10 (1), p. 1197.

Cheung, H. W. *et al.* (2011) 'Systematic investigation of genetic vulnerabilities across cancer cell lines reveals lineage-specific dependencies in ovarian cancer', *Proceedings of the National Academy of Sciences of the United States of America*, 108 (30), pp. 12372–12377.

Croft, D. *et al.* (2014) 'The Reactome pathway knowledgebase', *Nucleic acids research*, 42 (Database issue), pp. D472–7.

D'haeseleer, P., Liang, S. and Somogyi, R. (2000) 'Genetic network inference: from co-expression clustering to reverse engineering', *Bioinformatics* , 16 (8), pp. 707–726.

Dixit, A. *et al.* (2016) 'Perturb-Seq: Dissecting Molecular Circuits with Scalable Single-Cell RNA Profiling of Pooled Genetic Screens', *Cell*, 167 (7), pp. 1853–1866.e17.

Donovan, J. C., Milic, A. and Slingerland, J. M. (2001) 'Constitutive MEK/MAPK activation leads to p27 (Kip1) deregulation and antiestrogen resistance in human breast cancer cells', *The Journal of biological chemistry*, 276 (44), pp. 40888–40895.

Eraslan, G. *et al.* (2019) 'Single-cell RNA-seq denoising using a deep count autoencoder', *Nature communications*, 10 (1), p. 390.

Esteva, A. *et al.* (2017) 'Dermatologist-level classification of skin cancer with deep neural networks', *Nature*, 542 (7639), pp. 115–118.

Fitzgerald, J. B. *et al.* (2006) 'Systems biology and combination therapy in the quest for clinical efficacy', *Nature chemical biology*, 2 (9), pp. 458–466.

Frei, A. P. *et al.* (2016) 'Highly multiplexed simultaneous detection of RNAs and proteins in single cells', *Nature methods*, 13 (3), pp. 269–275.

Fröhlich, F. *et al.* (2017) 'Scalable Parameter Estimation for Genome-Scale Biochemical Reaction Networks', *PLoS computational biology*, 13 (1), p. e1005331.

Fröhlich, F. *et al.* (2018) 'Efficient Parameter Estimation Enables the Prediction of Drug Response Using a Mechanistic Pan-Cancer Pathway Model', *Cell systems*, 7 (6), pp.

567–579.e6.

Gardner, T. S. *et al.* (2003) ‘Inferring genetic networks and identifying compound mode of action via expression profiling’, *Science*, 301 (5629), pp. 102–105.

Garraway, L. A. and Jänne, P. A. (2012) ‘Circumventing cancer drug resistance in the era of personalized medicine’, *Cancer discovery*, 2 (3), pp. 214–226.

Gunning, D. and Aha, D. (2019) ‘DARPA’s Explainable Artificial Intelligence (XAI) Program’, *AI Magazine*, pp. 44–58. doi: 10.1609/aimag.v40i2.2850.

Gupta, S. and Davis, R. J. (1994) ‘MAP kinase binds to the NH2-terminal activation domain of c-Myc’, *FEBS letters*, 353 (3), pp. 281–285.

Hill, S. M. *et al.* (2017) ‘Context Specificity in Causal Signaling Networks Revealed by Phosphoprotein Profiling’, *Cell systems*, 4 (1), pp. 73–83.e10.

Hou, L. *et al.* (2016) ‘Patch-based Convolutional Neural Network for Whole Slide Tissue Image Classification’, *Proceedings / CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2424–2433.

Hug, S. *et al.* (2013) ‘High-dimensional Bayesian parameter estimation: case study for a model of JAK2/STAT5 signaling’, *Mathematical biosciences*, 246 (2), pp. 293–304.

Kingma, D. P. and Ba, J. (2014) ‘Adam: A Method for Stochastic Optimization’. Available at: http://arxiv.org/abs/1412.6980 (Accessed: 9 May 2019).

Klinger, B. *et al.* (2013) ‘Network quantification of EGFR signaling unveils potential for targeted combination therapy’, *Molecular systems biology*, 9, p. 673.

Korkut, A. *et al.* (2015) ‘Perturbation biology nominates upstream-downstream drug combinations in RAF inhibitor resistant melanoma cells’, *eLife*, 4. doi: 10.7554/eLife.04640.

Lamb, J. *et al.* (2006) ‘The Connectivity Map: using gene-expression signatures to connect small molecules, genes, and disease’, *Science*, 313 (5795), pp. 1929–1935.

Lei, W., Liu, F. and Ness, S. A. (2005) ‘Positive and negative regulation of c-Myb by cyclin D1, cyclin-dependent kinases, and p27 Kip1’, *Blood*, 105 (10), pp. 3855–3861.

Lezon, T. R. *et al.* (2006) ‘Using the principle of entropy maximization to infer genetic interaction networks from gene expression patterns’, *Proceedings of the National Academy of Sciences of the United States of America*, 103 (50), pp. 19033–19038.

Liang, J. and Liu, R. (2015) ‘Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network’, *2015 8th International Congress on Image*

and Signal Processing (CISP). doi: 10.1109/cisp.2015.7407967.

Lipton, Z. C. (2018) 'The mythos of model interpretability', *Communications of the ACM*, pp. 36–43. doi: 10.1145/3233231.

Locasale, J. W. and Wolf-Yadlin, A. (2009) 'Maximum entropy reconstructions of dynamic signaling networks from quantitative proteomics data', *PloS one*, 4 (8), p. e6522.

Lopez, R. *et al.* (2018) 'Deep generative modeling for single-cell transcriptomics', *Nature methods*, 15 (12), pp. 1053–1058.

Lotfollahi, M., Wolf, F. A. and Theis, F. J. (2019) 'scGen predicts single-cell perturbation responses', *Nature methods*, 16 (8), pp. 715–721.

Luna, A. *et al.* (2016) 'PaxtoolsR: pathway analysis in R using Pathway Commons', *Bioinformatics*, pp. 1262–1264. doi: 10.1093/bioinformatics/btv733.

Maglott, D. *et al.* (2011) 'Entrez Gene: gene-centered information at NCBI', *Nucleic acids research*, 39 (Database issue), pp. D52–7.

Mansoori, B. *et al.* (2017) 'The Different Mechanisms of Cancer Drug Resistance: A Brief Review', *Advanced pharmaceutical bulletin*, 7 (3), pp. 339–348.

Martín Abadi *et al.* (2015) 'TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems'. Available at: https://www.tensorflow.org/.

McDonald, E. R., 3rd *et al.* (2017) 'Project DRIVE: A Compendium of Cancer Dependencies and Synthetic Lethal Relationships Uncovered by Large-Scale, Deep RNAi Screening', *Cell*, 170 (3), pp. 577–592.e10.

Meinshausen, N. and Bühlmann, P. (2010) 'Stability selection', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, pp. 417–473. doi: 10.1111/j.1467-9868.2010.00740.x.

von Mering, C. *et al.* (2005) 'STRING: known and predicted protein-protein associations, integrated and transferred across organisms', *Nucleic acids research*, 33 (Database issue), pp. D433–7.

Meyer, P. E., Lafitte, F. and Bontempi, G. (2008) 'minet: A R/Bioconductor package for inferring large transcriptional networks using mutual information', *BMC bioinformatics*, 9, p. 461.

Mimitou, E. P. *et al.* (2019) 'Multiplexed detection of proteins, transcriptomes, clonotypes and CRISPR perturbations in single cells', *Nature methods*, 16 (5), pp. 409–412.

Mokhtari, R. B. *et al.* (2017) ‘Combination therapy in combating cancer’, *Oncotarget*. doi: 10.18632/oncotarget.16723.

Molinelli, E. J. *et al.* (2013) ‘Perturbation biology: inferring signaling networks in cellular systems’, *PLoS computational biology*, 9 (12), p. e1003290.

Montavon, G., Samek, W. and Müller, K.-R. (2018) ‘Methods for interpreting and understanding deep neural networks’, *Digital Signal Processing*, pp. 1–15. doi: 10.1016/j.dsp.2017.10.011.

Nelander, S. *et al.* (2008) ‘Models from experiments: combinatorial drug perturbations of cancer cells’, *Molecular systems biology*, 4, p. 216.

Niepel, M. *et al.* (2017) ‘Common and cell-type specific responses to anti-cancer drugs revealed by high throughput transcript profiling’, *Nature communications*, 8 (1), p. 1186.

Norman, T. M. *et al.* (2019) ‘Exploring genetic interaction manifolds constructed from rich phenotypes’, *bioRxiv*. doi: 10.1101/601096.

Nyman, E. *et al.* (2019) ‘Perturbation biology links temporal protein changes to drug responses in a melanoma cell line’. doi: 10.1101/568758.

Osaki, L. H. and Gama, P. (2013) ‘MAPK signaling pathway regulates p27 phosphorylation at threonin 187 as part of the mechanism triggered by early-weaning to induce cell proliferation in rat gastric mucosa’, *PloS one*, 8 (6), p. e66651.

Ryall, K. A. and Tan, A. C. (2015) ‘Systems biology approaches for advancing the discovery of effective drug combinations’, *Journal of cheminformatics*, 7, p. 7.

Schmiester, L. *et al.* (2020) ‘Efficient parameterization of large-scale dynamic models based on relative measurements’, *Bioinformatics* , 36 (2), pp. 594–602.

Schraivogel, D. *et al.* (2020) ‘Targeted Perturb-seq enables genome-scale genetic screens in single cells’, *Nature methods*, 17 (6), pp. 629–635.

Sears, R. (2000) ‘Multiple Ras-dependent phosphorylation pathways regulate Myc protein stability’, *Genes & Development*, pp. 2501–2514. doi: 10.1101/gad.836800.

Şenbabaoğlu, Y. *et al.* (2016) ‘A Multi-Method Approach for Proteomic Network Inference in 11 Human Cancers’, *PLoS computational biology*, 12 (2), p. e1004765.

Stapor, P. *et al.* (2019) ‘Mini-batch optimization enables training of ODE models on large-scale datasets’. doi: 10.1101/859884.

Süli, E. and Mayers, D. F. (2003) *An Introduction to Numerical Analysis*. Cambridge University Press.

Thelisson, E. (2017) ‘Towards Trust, Transparency and Liability in AI / AS systems’, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. doi: 10.24963/ijcai.2017/767.

Tibes, R. *et al.* (2006) ‘Reverse phase protein array: validation of a novel proteomic technology and utility for analysis of primary leukemia specimens and hematopoietic stem cells’, *Molecular cancer therapeutics*, 5 (10), pp. 2512–2521.

Tsherniak, A. *et al.* (2017) ‘Defining a Cancer Dependency Map’, *Cell*, 170 (3), pp. 564–576.e16.

Vanhaelen, Q., Aliper, A. M. and Zhavoronkov, A. (2017) ‘A comparative review of computational methods for pathway perturbation analysis: dynamical and topological perspectives’, *Molecular bioSystems*, 13 (9), pp. 1692–1704.

Wang, K. *et al.* (2009) ‘Genome-wide identification of post-translational modulators of transcription factor activity in human B cells’, *Nature biotechnology*, 27 (9), pp. 829–839.

Wang, T. *et al.* (2014) ‘Genetic screens in human cells using the CRISPR-Cas9 system’, *Science*, 343 (6166), pp. 80–84.

Wroblewska, A. *et al.* (2018) ‘Protein Barcodes Enable High-Dimensional Single-Cell CRISPR Screens’, *Cell*, 175 (4), pp. 1141–1155.e16.

Wrzodek, C. *et al.* (2013) ‘Precise generation of systems biology models from KEGG pathways’, *BMC systems biology*, 7, p. 15.

Yi, S. *et al.* (2017) ‘Functional variomics and network perturbation: connecting genotype to phenotype in cancer’, *Nature reviews. Genetics*, 18 (7), pp. 395–410.

Zaal, E. A. and Berkers, C. R. (2018) ‘The Influence of Metabolism on Drug Response in Cancer’, *Frontiers in Oncology*. doi: 10.3389/fonc.2018.00500.

Zhou, J. and Troyanskaya, O. G. (2015) ‘Predicting effects of noncoding variants with deep learning-based sequence model’, *Nature methods*, 12 (10), pp. 931–934.

Zou, M. and Conzen, S. D. (2005) ‘A new dynamic Bayesian network (DBN) approach for identifying gene regulatory networks from time course microarray data’, *Bioinformatics* , 21 (1), pp. 71–79.