ARTICLES
https://doi.org/10.1038/s42256-021-00357-4
nature machine intelligence

Deep learning prediction of patient response time course from early data via neural-pharmacokinetic/pharmacodynamic modelling

James Lu[1]✉, Brendan Bender[1], Jin Y. Jin[1]✉ and Yuanfang Guan[2]

Longitudinal analyses of patient response time courses following doses of therapeutics are currently performed using pharmacokinetic/pharmacodynamic (PK/PD) methodologies, which require considerable human experience and expertise in the modelling of dynamical systems. By utilizing recent advancements in deep learning, we show that the governing differential equations can be learned directly from longitudinal patient data. In particular, we propose a novel neural-PK/PD framework that combines key pharmacological principles with neural ordinary differential equations. We applied it to an analysis of drug concentration and platelet response from a clinical dataset consisting of over 600 patients. We show that the neural-PK/PD model improves on a state-of-the-art model with respect to metrics for temporal prediction. Furthermore, by incorporating key PK/PD concepts into its architecture, the model can generalize and enable the simulations of patient responses to untested dosing regimens. These results demonstrate the potential of neural-PK/PD for automated predictive analytics of patient response time course.

Longitudinal analysis of patient response time course following doses of therapeutics is an important topic for drug development and personalized medicine; however, this is a highly challenging task due to the variability of patient response to treatments and the complexity of drug mechanisms. In response to the need for better characterizing the dynamics between dosing, drug concentration and patient response, the pharmacokinetic/pharmacodynamic (PK/PD) discipline has been developed to link systemic drug concentration kinetics (that is, PK) to the resulting pharmacological effects over time (that is, PD) using mathematical models[1]. These models enable the description and prediction of the time course of physiological effects (for example, tumour size, platelet count and so on) in response to drugs of various dosage regimens[1,2]. Pharmacokinetic/pharmacodynamic models are typically developed using ordinary differential equations (ODEs), the construction of which have relied on human modellers' abstraction of data into dynamical systems. The current state-of-the-art methodology for estimating the set of model parameters in PK/PD models is the population approach[3], whereby certain statistical distributions of parameters and error models are assumed, and iterative optimization techniques are subsequently applied to computationally minimize the discrepancy between an observed and predicted trajectory in some appropriate error metric[2,4,5]. The performance of alternative models is compared and a selection is ultimately made on the basis of various diagnostic criteria and modelling statistics[4,5]. As described above, the population-PK/PD (pop-PK/PD) modelling paradigm is highly iterative in nature. The accuracy of such models for making temporal predictions depends on the modeller's expertise and ability to describe complex datasets with mathematical equations; however, as the range of data modalities in modern biomedical applications increase to include imaging, high dimensional assays and continuous monitoring devices, it becomes ever more challenging for human modellers to glean insight from such large volumes of data. Coupled to the growing data is the need to improve the ability of PK/PD models to perform temporal extrapolations, which is key to precision dosing applications[6]. Motivated by these challenges, we explore the possibility of using deep learning to build PK/PD models to augment human capabilities in abstracting dynamical systems from patient data following drug treatment while enhancing predictive accuracy.

With the recent development of the neural-ODE methodology[7], it has become possible to consider an ODE modelling paradigm whereby one can learn the governing equations algorithmically and directly from the data[8,9]. In particular, this novel machine learning approach generates the input-to-output mapping as the numerical integration of an ODE system described by a neural network, in which the backpropagation is carried out by an adjoint solution method[7]. The neural-ODE methodology is well suited for time-series analysis and especially in the PK/PD setting, as both the dosing and measurement times can be irregular. In particular, it has been applied to fields such as the life sciences[10], image processing[11] and computational physics[12–14]. Although there are several examples of neural-ODE methodologies that have been developed and applied to publicly available biomedical datasets (such as PhysioNet)[10,15,16], there is currently no implementation in which the dosing of a therapeutic drug and its concentration data are both explicitly represented in the model. In addition, as these model formulations do not place constraints on the form of ODE systems being constructed and lack the incorporation of key pharmacological principles into their architecture, their ability to generalize from the training set and predict unseen dosing regimens remain in question.

In this work we propose a new deep learning approach to build PK/PD models that directly learn the governing equations from data with the aim of predicting patient response time course as well as being able to simulate the effects of unseen dosing regimens. Our approach is novel in two ways. First, the architecture of our deep learning model ensures that the pharmacological principle of dose-concentration-effect is preserved; that is, the model assumes

[1]Department of Clinical Pharmacology, Genentech, South San Francisco, CA, USA. [2]Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, USA. ✉e-mail: lu.james@gene.com; jin.jin@gene.com

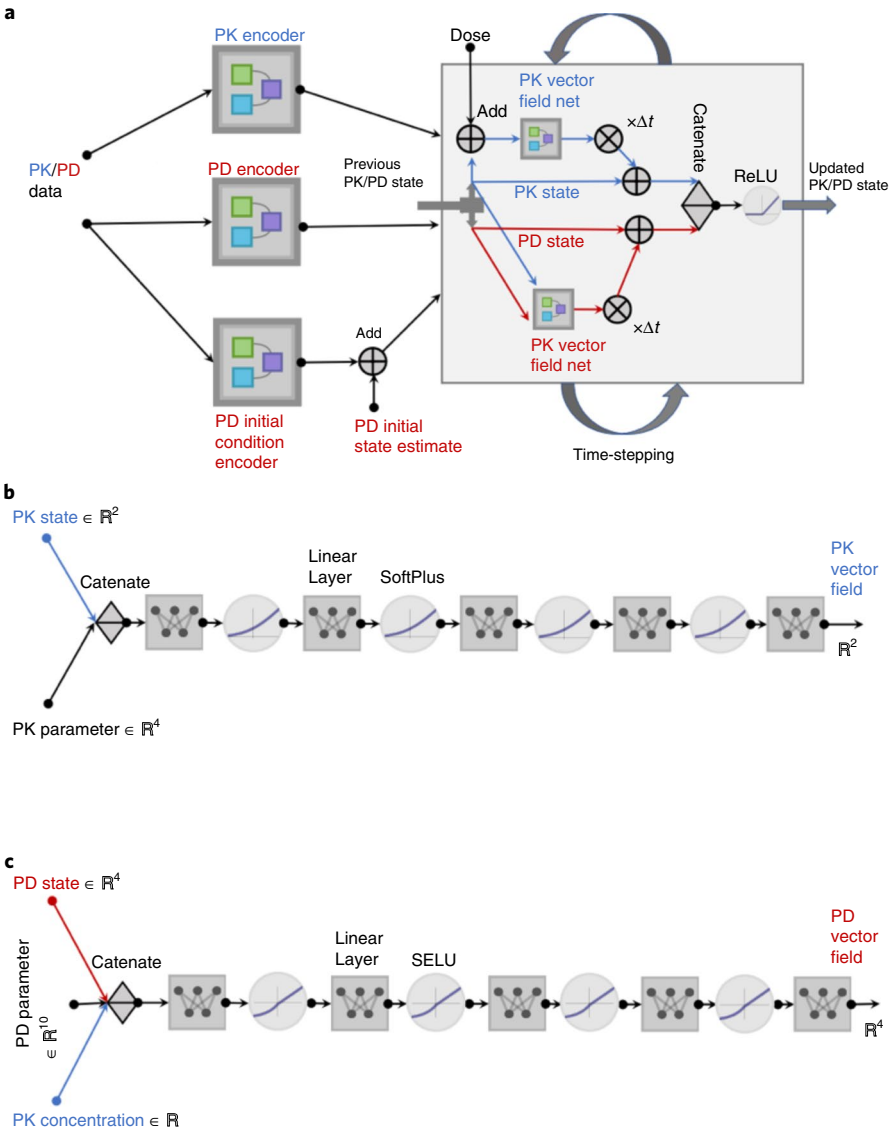


Fig. 1 | Schematic diagram of the neural-PK/PD model. a, The input PK/PD data (consisting of the measured data and dosages versus time) is passed as a table of numeric values into the neural network, which is subsequently encoded into vectors of various dimensions by the three encoders (PK, PD and PD initial condition). The encoded vectors are fed into a recurrent neural network, which unfolds along the time dimension and, together with the input doses, produces a sequence of predictions for the PK and PD states. **b**, The PK vector field net consists of linear and SoftPlus[33] layers, which takes the current value of the PK state as well the output of the PK encoder as inputs. **c**, The PD vector field net consists of linear and SELU[37] layers, which takes the current value of the PD state, the output of the PD encoder as well as the PK concentration as inputs.

the causal relationship of dosing driving the drug concentration, which in turn drives the effect dynamics. Second, the network architecture is constructed in a manner such that the dosing data enters the model via not one but two ports, thereby ensuring the ability of the model to not only predict the existing treatment data but also enable the simulation of what-if scenarios whereby the patients' dosing regimens are modified. Our work illustrates how by incorporating key domain specific modelling principles into the neural network architecture, human and machine intelligence can work together in building dynamical systems that are more likely to generalize well beyond the training data and be better embraced by the domain experts.

Herein we detail a sequential methodology for the construction of neural-PK and neural-PK/PD models. We demonstrate the approach in describing and predicting drug concentration and platelet dynamics following treatment with trastuzumab emtansine (T-DM1), an approved anti-cancer therapy (intravenous administration at 3.6 mg kg^{-1} once every three weeks or Q3W) for the treatment of human epidermal growth factor receptor 2 (HER2)-positive metastatic breast cancer in patients failing treatment beforehand with trastuzumab and a taxane[17]. Thrombocytopenia—a decrease in platelets that requires dose reductions and delays—is the dose-limiting toxicity for T-DM1[17]. To show the utility of this methodology, we benchmark against the original pop-PK/PD analyses[17,18], which utilized the myelosuppression PKPD modelling approach—considered to be the gold-standard for modelling such clinical data—proposed by Friberg and colleagues[19]. We develop a neural network to search within a space of ODE systems no larger in dimension than that of the existing pop-PK/PD model[17] and compare the resulting neural-PK/PD model with pop-PK/PD in terms of their ability to predict future platelet counts from early data at the individual patient level. Although existing pop-PK models

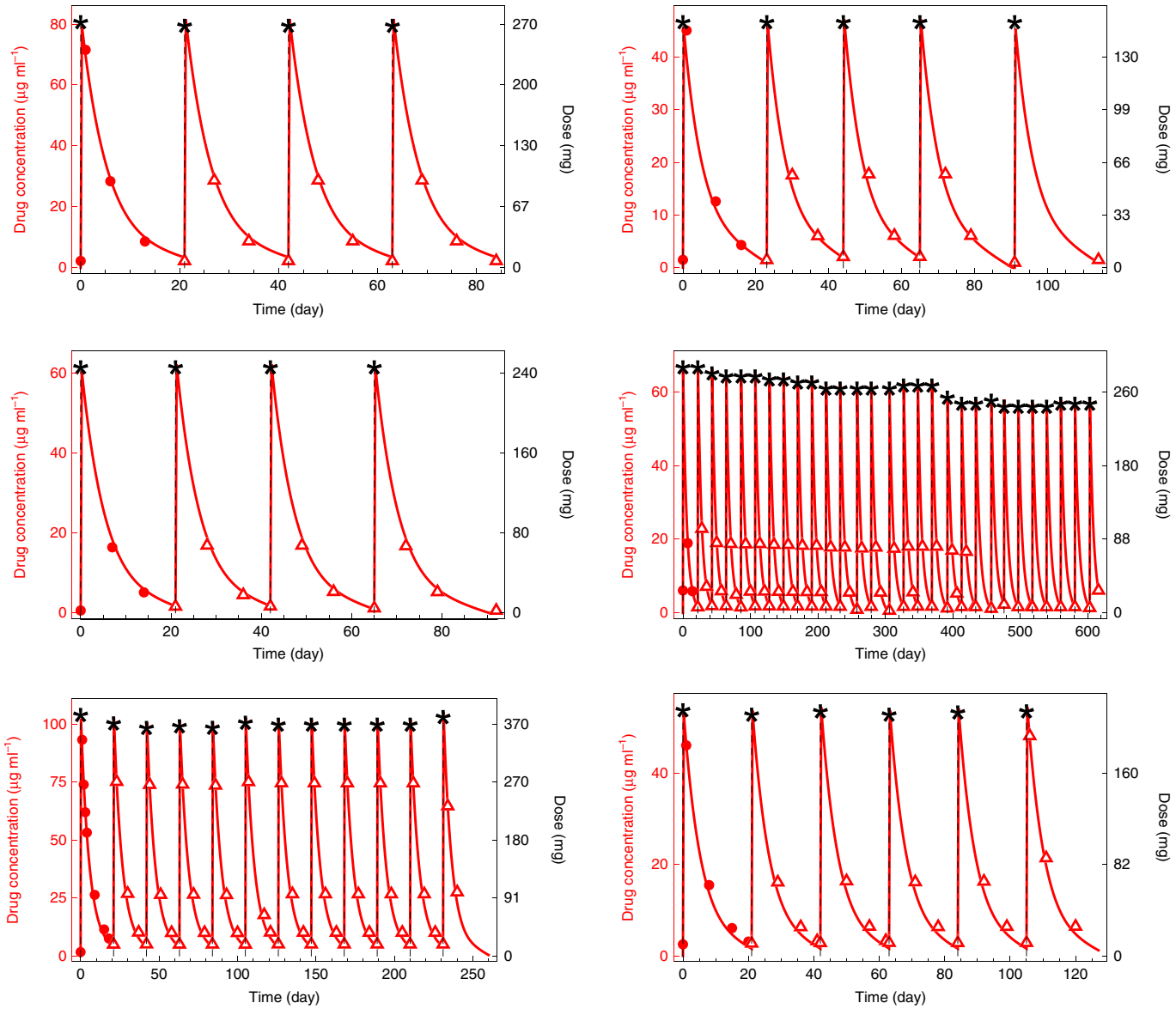


Fig. 2 | An illustration of the ability of neural-PK model to predict complete PK profiles from having being shown only the early PK data from the first dosing cycle. Results for six randomly selected test patients are shown. Pharmacokinetic data within the observation time window $t \in [0, 21]$ is passed onto the PK encoder, and the model predicts the PK time course given the complete dosing data. The black stars are the dosages, the filled red circles are the observed drug concentration data for the first dosing cycle ($t < 21$ days), the open red triangles are unobserved drug concentrations for $t \geq 21$ days and solid red curves represent the neural-PK model predictions.

perform well for PK predictions, PD data are intrinsically much more noisier and difficult to predict, hence PD prediction is the focus of our work. Finally, we illustrate the generalizability of neural-PK/PD predictions by performing simulations to predict the effects of alternate (and untested) dosing regimens.

Results

Overview of model. We incorporate the basic principles of PK/PD[1] into our neural-PK/PD architecture (Fig. 1) by encoding the following standard PK/PD assumptions: first, the dynamics of PK is driven by dosing and is independent of PD; second, the dynamics of PD is driven by PK as well as by itself[1]. We built the above assumptions into the computational graph of the model via the explicit representation of the dose input into the ODE submodule as shown in Fig. 1a, and in the dependence of the PK and PD vector fields shown in Fig. 1b and Fig. 1c. In particular, the dose data enters as Dirac delta forcing to the PK component of the ODE system (refer to the Methods and Extended Data Fig. 1b for further details).

PK (drug concentration) data is recapitulated almost perfectly. In the current work, the PK and PD parts of the network are built in a sequential manner[20], as is the case for the original pop-PK/PD model[17,18] that we benchmark against. In particular, we take a pop-PK model that was built previously for T-DM1[21,22] and train a neural-PK model to mimic the pop-PK model by reading in the early PK data and predicting the future time points. We used a T-DM1 treatment dataset involving 665 patients, with a median observation and dosing record of 169 days (minimum, 1 day; maximum, 862 days). We split the total number of available patients into a training and testing set (please refer to the Methods for details on the training and the structure of the neural-PK model). As illustrated in Fig. 2, the trained neural-PK model can effectively predict

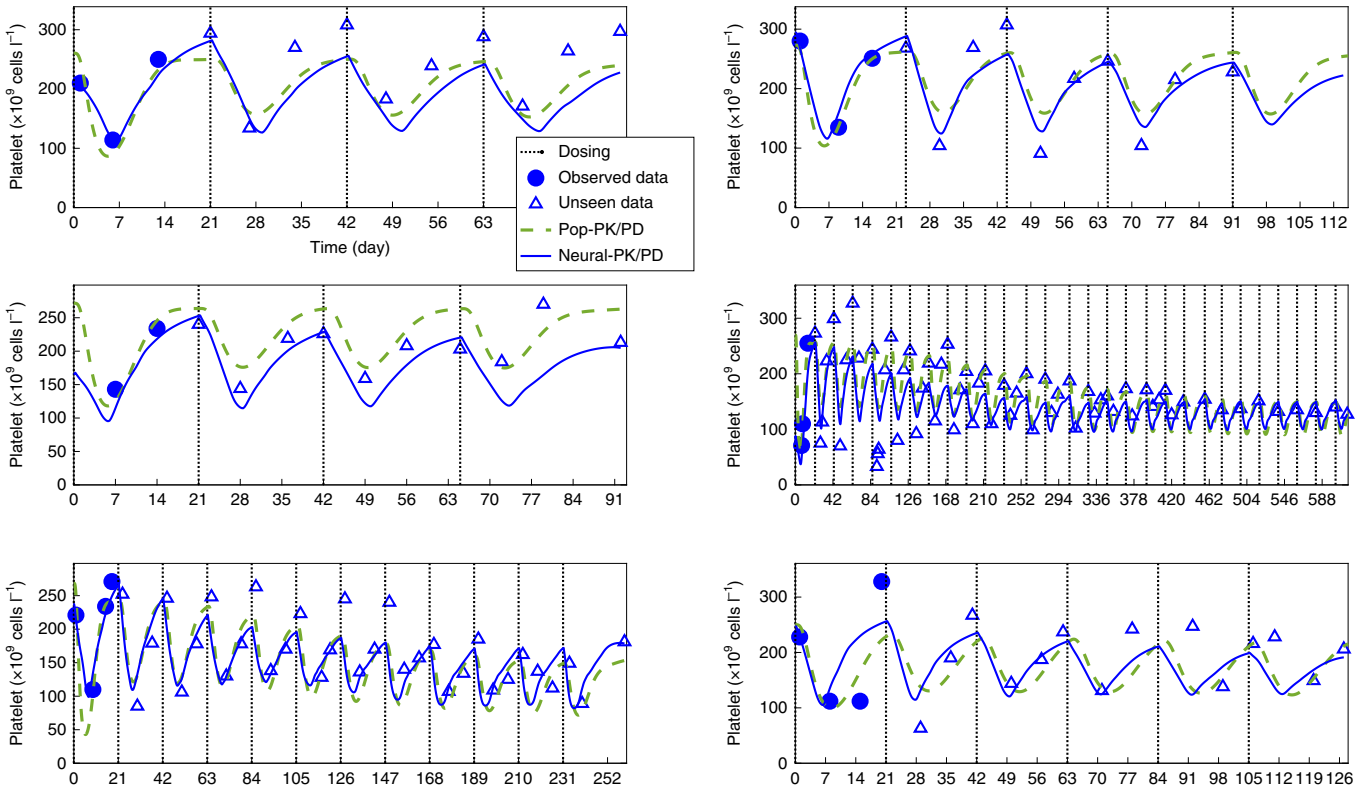


Fig. 3 | Comparisons of platelet predictions from pop-PK/PD model versus neural-PK/PD model in selected patients from the test set, whereby only data from $t < 21$ days is observed. The plots show similar qualitative dynamics between the two models, as well as some quantitative differences in their predictions.

(on unseen test patients) the complete time course of the pop-PK model for T-DM1 drug concentration by using only the observed PK data up to day 21, which is the first cycle of treatment. For a quantitative assessment of the neural-PK model, we evaluated it on the $n = 133$ test patients with 3,228 PK observations made from times of $t \geq 21$ days. The predicted drug concentrations from the neural-PK model correspond well with the predicted drug concentrations from the original pop-PK model, with an R^2 of 0.98, a correlation coefficient of 0.99, and an r.m.s.e. of 2.67.

Qualitative dynamics of PD (drug response) is recapitulated. By using the trained neural-PK submodule to drive the neural-PD submodule, we train the neural network to generate a dynamical system that recapitulates the observed patient platelet trajectories (see the Methods for details). Figure 3 shows a comparison between predictions from the neural-PK/PD model and the pop-PK/PD model[17] with the actual platelet data for randomly selected patients from the test set. In particular, the observed drug concentration and platelet data from the first dosing cycle (that is, up to 21 days) are provided to the neural network, which subsequently generates the predicted drug concentration and platelet response for all future time points (after 21 days) on the basis of the individual dosing information. The result shows that by incorporating PK/PD principles into its network architecture, but otherwise without additional human input, the proposed methodology is able to generate a PK/PD model that demonstrates comparable qualitative behaviour (in the sense of dynamical characteristics of the system) with the state-of-the-art model[17]. In particular, as shown in Fig. 3, the neural-PK/PD model predictions demonstrate a drop in the platelet count post dosing, the subsequent recovery dynamics, as well as a gradual decrease in both the peaks and nadirs of platelet counts in some patients who have been treated for long durations. Although considerable human expertise and effort was required to build a complex PK/PD model[17] (containing negative feedback, time-dependent and nonlinear terms) based on the Friberg formalism[19], a qualitatively similar dynamical system was constructed in an automated fashion by applying neural-PK/PD modelling directly to data.

Neural-PK/PD model outperforms the current gold-standard. We benchmark our neural-PK/PD model against the current gold-standard pop-PK/PD model[17]. We considered the following three sets of observation limits: $t_{obs} = 21$, 42 and 63 days. In each case, we use the data within the initial observation window ($t < t_{obs}$) to predict all of the future platelet data ($t \geq t_{obs}$). Note that we use non-empty observation windows as the model predictions would default to the population average if no data were used, which would be a poor prediction for individual patients given the interindividual variability present in the data[2,4]. Figure 4 presents a comparison of pop-PK/PD and neural-PK/PD model predictions versus the observed data (ground truth), using both the R^2 and r.m.s.e. to demonstrate that the latter surpasses the former. In fact, as shown in Table 1 (cases a–c), the performance of neural-PK/PD surpasses that of pop-PK/PD by a sizeable margin for all observation windows (obs = 21, 42 and 63 days). We believe the gain in predictive performance primarily arises from the neural-PK/PD formulation effectively searching over a broader space of equation systems as compared with the space of algebraic expressions that human modellers work with.

What are the potential implications of improving the prediction accuracy using neural-PK/PD modelling as compared with the current pop-PK/PD modelling approaches? We considered the following scenario: suppose we wish to predict the individual platelet counts in test patients over $t \geq 42$ days. Using the existing pop-PK/PD approach whereby data is observed for $t_{obs} < 42$ days, we obtain

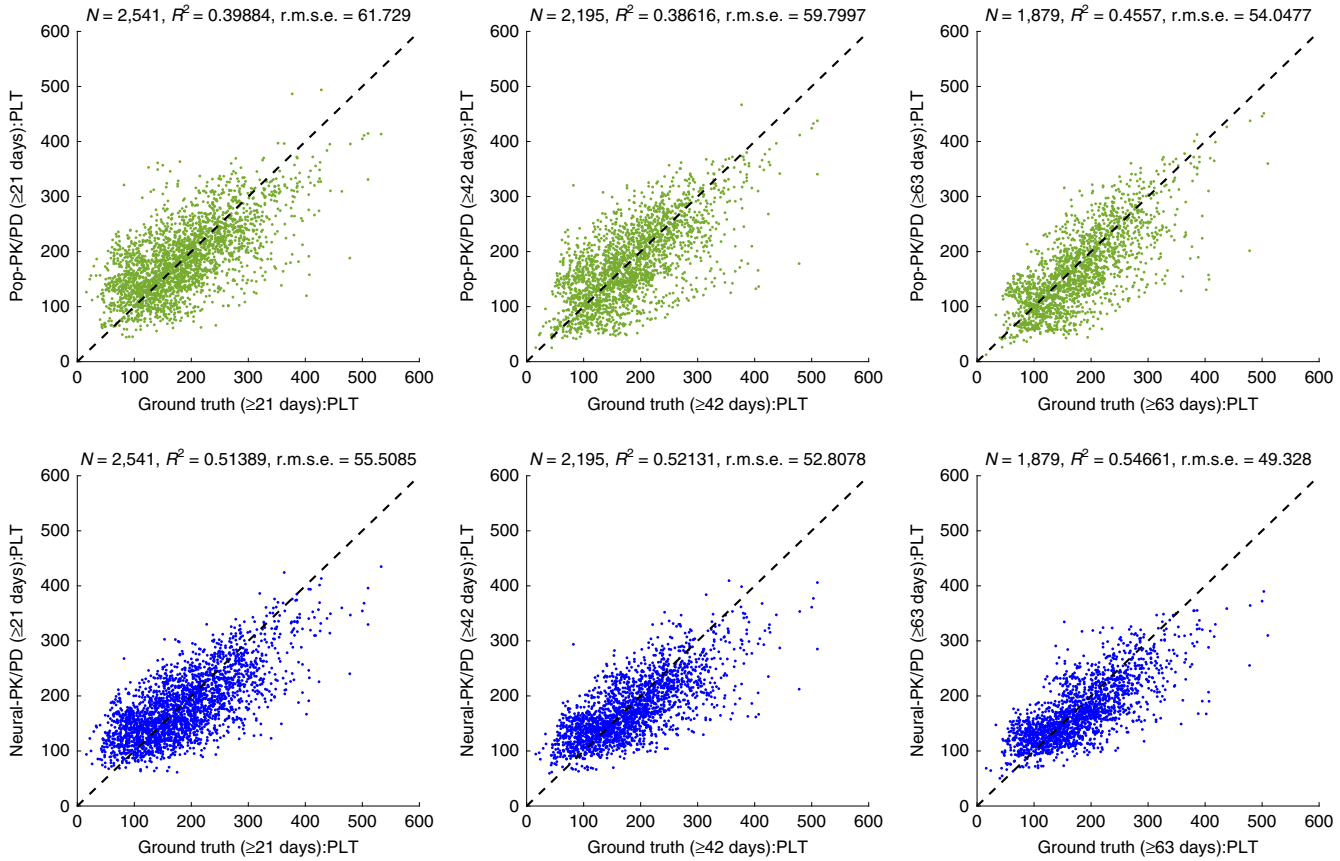


Fig. 4 | Comparisons of the model prediction versus ground truth using the pop-PK/PD and neural-PK/PD models. We consider the scenarios of observing platelet (PLT) data within treatment cycles 1–3 (that is, corresponding to time <21, 42 and 63 days, respectively) and attempting to predict the future platelet time course (that is, corresponding to $t \geq 21$, 42 and 63 days respectively). In each case, *N* refers to the number of predictions made for the corresponding scenario. The prediction results show that neural-PK/PD model has a numerically higher R^2 and lower r.m.s.e. in predicting future platelet counts than the pop-PK/PD model.

Table 1 | Comparisons of predictive performance for pop-PK/PD versus neural-PK/PD, shown as mean ±s.d. over five bootstrapped samples

Case	Observation		Prediction		Pop-PK/PD		Neural-PK/PD	
	Window (day)	No. of obs.	Window (day)	No. of pred.	R^2	r.m.s.e.	R^2	r.m.s.e.
a	$0 < t < 21$	413	$t \geq 21$	2,541	0.40 ± 0.03	62 ± 1	**0.51 ± 0.02**	**56 ± 1**
b	$0 < t < 42$	759	$t \geq 42$	2,195	0.39 ± 0.02	60 ± 1	**0.52 ± 0.01**	**53 ± 1**
c	$0 < t < 63$	1,075	$t \geq 63$	1,879	0.46 ± 0.02	54 ± 1	**0.55 ± 0.02**	**49 ± 2**
d	$0 < t < 21$	413	$t \geq 42$	2,195	-	-	**0.45 ± 0.02**	**56 ± 1**

Cases a–c compare pop-PK/PD with neural-PK/PD on the basis of identical sets of observation and prediction data; the results demonstrate a superior performance for the latter methodology. In case d, neural-PK/PD predictions using 413 early observations (from $0 < t < 21$ days) are shown to give rise to numerically higher performance than pop-PK/PD (case b), which utilizes nearly double the number of observations (759) from the observation window $0 < t < 42$ days. In case d, the pop-PK/PD model result was not relevant for the comparison being made and hence is not shown.

an R^2 of 0.39 and r.m.s.e. of 60 (refer to case b in Table 1). By contrast, using the proposed neural-PK/PD approach we can make more precise predictions ($R^2 = 0.45$, r.m.s.e. = 56.36) using only half the observation window ($t_{obs} < 21$ days; refer to case d in Table 1). The ability of the neural-PK/PD model to use less data while improving the forecast precision for patients' future response as compared with the current state-of-the-art model makes it a promising methodology that could potentially enable more robust predictive analytics based on earlier observation.

Simulating alternative dosing regimen. One key use of pop-PK/PD model is to perform simulations of new dosing regimens of interest[4,5] and predict the corresponding drug concentration and response of such untested scenarios in patients. By design, dosing enters the neural-PK/PD model as an independent input, thereby ensuring that different dosing schedules can be evaluated as desired. As a demonstration of the utility of neural-PK/PD models to predict alternative dosing regimens, we show in Fig. 5 the model simulations for Q3W dosing at 3.6 mg kg^{-1}, Q1W dosing at 2.4 mg kg^{-1}, as well as dosing once every three days (Q3D) at 1.0 mg kg^{-1} of T-DM1, for the patients in the training set. Note that the majority of the patients in the training set were treated through Q3W dosing, with the exception of a handful of patients who were treated through Q1W. In particular, no patient was given Q3D and thus it is a dose schedule that the model has never been trained on. The dosage of 1.0 mg kg^{-1} for Q3D was chosen to be

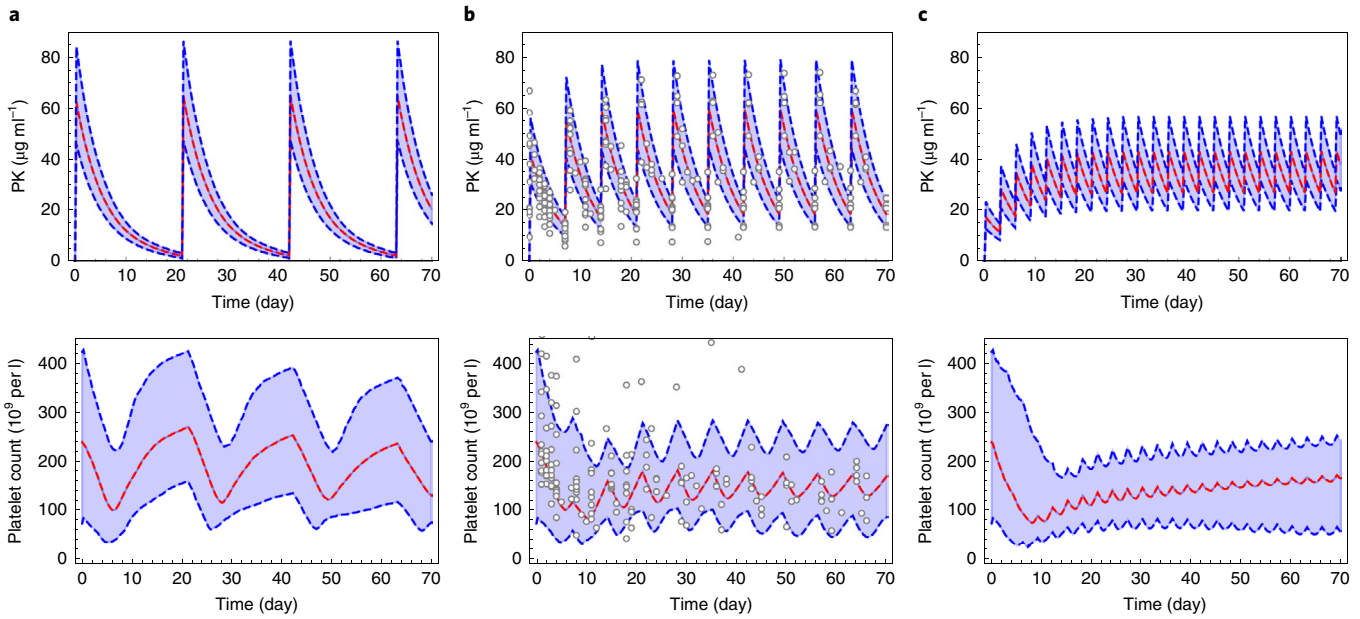


Fig. 5 | Neural-PK/PD performing population simulations using $n = 532$ training patients. a–c, Q3W dosing at 3.6 mg kg^{-1} of T-DM1 (**a**); Q1W dosing at 2.4 mg kg^{-1} of T-DM1 (**b**); Q3D dosing at 1.0 mg kg^{-1} of T-DM1 (**c**). The dashed red lines show the population medians, whereas the dashed blue lines show the 5th to 95th percentiles of the training patients. In **b**, we have superimposed the data from the 13 patients who were treated with Q1W at 2.4 mg kg^{-1} of T-DM1 onto the model predictions, demonstrating the model's ability to generalize from the Q3W regimen. Furthermore, despite not having any patient given Q3D in the training set, the model was able to generate realistic profiles that demonstrate the expected drug concentration (PK) accumulation and frequent platelet fluctuations as a result of the proposed model architecture which is based upon PK/PD principles.

equivalent to the 2.4 mg kg^{-1} Q1W based on the weekly total dose (that is, $2.4 \times \frac{3}{7} = 1.029$). The simulation results shown in Fig. 5 demonstrate a larger peak-to-nadir swing of platelet counts in Q3W than Q1W, but with comparable nadir values, a finding that is consistent with the pop-PK/PD simulation results[17]. Furthermore, the 2.4 mg kg^{-1} Q1W simulation result compares well with data from 13 patients who were treated with this dose and regimen. Finally, the PK accumulation observed for Q3D is consistent with the expected PK behaviour of T-DM1. The ability of the model to generate simulations with previously unseen dosing stems from the neural network architecture having been designed based on PK/PD principles, resulting in meaningful extrapolations.

Discussion

In this work we present a novel application of a deep learning approach to predict drug concentration and response time course, for the purpose of enabling personalized predictions in individual patients and predicting the effects of different dose schedules. Our work provides a head-to-head comparison of a pop-PK/PD model with a deep learning model. As distinct methodologies analysing drug concentration/response data and making clinical inferences, the traditional pop-PK/PD and the proposed novel neural-PK/PD approaches have their respective pros and cons. Although the former approach incorporates basic pharmacological principles that enable meaningful interpretation, model development can be time consuming due to the need for human-driven testing of alternative models and various diagnostic evaluations, thus hampering the ability to use it for real-time predictions or complex datasets. On the other hand, although neural-PK/PD is driven by data and can be generated algorithmically with little direct human supervision thus saving time and human resources, it remains to be validated on many datasets, and we hope this study will spur such analyses in the near future. Furthermore, the incorporation of variability into neural-PK/PD models to enable clinical trial simulations remains to be further developed.

Although deep learning models have been shown to have impressive ability to directly learn from vast amounts of data and enable predictions with little human intervention, existing techniques are known to have certain drawbacks[23–25]. First, many deep learning models tend to be data hungry[26] and their applicability to datasets of moderate size that arise from clinical trial settings may be limited. Second, it is known that deep learning models may use features in the training data that do not generalize well to other situations[24], thereby potentially lowering the utility of such models especially for prediction of untested scenarios. We believe that the incorporation of key domain specific concepts into deep learning frameworks is an effective way to combine human and machine intelligence and can be crucial for the successful applications of such models. Within the domain of longitudinal analysis of patient treatment data, we propose the incorporation of well-established PK/PD principles into neural-ODE framework[7] by an appropriate choice of network architecture as a way to ensure that these models would serve well for the purpose of individualized predictions of the effects of untested dosing regimen. We demonstrate the feasibility of such novel approach for drug concentration and response prediction based on a legacy clinical trial dataset consisting of more than 600 patients. Although the results shown in this work demonstrate the proposed neural-PK/PD modelling approach can improve the current gold-standard pop-PK/PD modelling on prediction metrics, the clinical implications and meaningfulness of such improvements need to be further assessed in future work. The importance of using machines to aid humans build models will increase as technological advancements lead to an ever-increasing set of diagnostic modalities and monitoring devices that generate growing amounts of patient treatment data in real time; relying on human insight alone to generate models that describe the observed data will become increasingly challenging in the digital age. With the joint requirements of automating modelling to save time and human resources while ensuring the pharmacological meaningfulness and the predictability of the generated models to inform dosing considerations, it is

important to bridge modern deep learning and traditional PK/PD methodologies. Although machine learning is currently not part of clinical pharmacology modelling applications, it is appreciated that a number of opportunities exist for leveraging both paradigms[27]. We propose that the "artificial intelligence (AI)-enabled clinical pharmacologists" of the future[28] would tap into neural-PK/PD as one of the advanced analytics tools to understand and predict drug concentration and response for dosing recommendation. Moreover, as neural-PK/PD models speak the language of neural networks, they could be combined with other deep learning models for novel data types and can also be integrated into digital devices for automated longitudinal patient monitoring to generate treatment insights.

Although the proposed neural-PK/PD formulation demonstrates encouraging results, there are several areas that remain topics for future work. The current model in the form of a recurrent architecture that takes fixed time steps in the numerical approximation of the underlying ODE system. Various efforts are underway to generalize the current framework by incorporating neural-ODE solvers that take adaptive step sizes (for example, ref. [7]) to account for the arbitrary time values that may be present in the dataset (that is, measurement and dosing times). Although the dataset shown in this work involves time points that take on integer values (in the unit of day) and hence directly applicable to the proposed model, the incorporation of adaptive step size is expected to provide increased efficiency and accuracy in the general setting. For this extension, although the recurrent neural net shown in Fig. 1 would need to be replaced by an adaptive ODE solver[7], the architecture of the proposed neural-PK/PD model would remain the same. Finally, in the current work we did not use any baseline features of the patients (such as demographics, pre-treatment laboratory values and so on) for a head-to-head comparison with the original pop-PK/PD model[17], which similarly did not include any covariate effects. Incorporation of the available baseline features may better characterize patient heterogeneity and improve the model predictivity. The neural-PK/PD model can incorporate baseline features in a number of ways, including being added to one of the input or decoder layers, or as a new input channel; however, a thorough investigation of the appropriate architecture remains a topic for future work. In general, this novel neural-PKPD approach also needs to be tested with more drugs and indications, as well as various types of therapeutic and adverse responses to further support the robustness and predictivity of the approach in a variety of clinical settings. As model interpretation is important for clinical applications, further research is also needed to shed light on explaining neural-PK/PD models. Deep learning models are not in the form of explicit algebraic expressions, nevertheless techniques such as saliency maps could be used to identify important input information that contribute to the prediction. Although individual parameters of a neural network do not have meaning on their own in contrast to pop-PK/PD models, aspects of the model could nevertheless be analysed including visualizing and quantitatively examining the vector fields. A related question is that of quantifying the level of confidence in the model predictions both at the individual level as well as at the population level, which remains an important topic for further research. To conclude, the promising results shown in this work and the potential advantages offered by this novel neural-PK/PD methodology warrants further development and testing. We expect that the development and deployment of neural-PK/PD models would help to facilitate clinical drug development and advance personalized medicine in clinical practice.

Methods

In this work, the platelet count measured in patients' blood corresponds to the PD variable whose time course we attempt to describe in relation to the PK (or drug concentration). We compare the neural-PK/PD model with pop-PK/PD in terms of their ability to predict future platelet counts from early data by taking 80% of the total patients in the data as the training set and the remaining 20% as the test set, in a manner following the machine learning paradigm[29,30]. In particular, we consider a hypothetical scenario whereby we have obtained data from a past clinical study (that is, the training set) and built a model on it from the complete time course; subsequently, we would like to apply the model to a new trial of a similar patient population (that is, the test set) and predict the individual patient future platelet time course from having observed only the early data. We thus aim to use the model to make future predictions on the test set based on their early data, as well as using all data from the training set. Although pop-PK/PD models are typically evaluated based on their ability to describe the available data with parsimony considerations[4,5] rather than for their ability to make temporal prediction, the current approach is one way to compare the models on an equal footing.

Model architecture. The precise treatment of dosing is a key part of our model implementation that follows the traditional PK/PD modelling paradigm and helps to ensure the generalizability of the model predictions to unseen dose schedules. The PK components of the ODE vector field are not influenced by the PD states, whereas the PD components of the ODE vector field are influenced by both the PK and PD states. The individual patient data—including the observed drug concentration, dosed amounts and pharmacodynamic response (that is, platelet count in the current example)—is fed into the network as input PK/PD data. An innovative feature of our model is that drug dosing enters the network at two locations: as part of the input PK/PD data, and also through the dose port. Although the dosing information that forms part of the PK/PD data represents what doses patients have been treated with, the dosing that enters the dose port represents what the model uses in making predictions. Thus, to enable both the modelling of treatment data as well as the simulation of alternate doses, we have dosing entering the model at two different locations. For the purpose of reproducing the data in the training set, the doses that enter the two separate ports are identical; however, once the model has been trained and a novel dosing regimen is to be tested, the dosing data that enters the dose port is modified as desired.

Data. The data consists of longitudinal platelet response from 665 patients receiving T-DM1, including patients from one phase I study (TDM3569g), three phase II studies (TDM4258g, TDM4374g and TDM4450g) and one phase III study (TDM4370g). Please refer to refs. [17,18,21] for further details regarding the clinical studies and the patient demographics. All patients provided written informed consent. The majority of the patient data are from the phase II/III studies, all of whom received T-DM1 at 3.6 mg kg^{-1} Q3W. Patients in the phase I study received various doses and schedules of T-DM1: 0.3 mg kg^{-1} Q3W ($n=3$), 0.6 mg kg^{-1} Q3W ($n=1$), 1.2 mg kg^{-1} Q3W ($n=1$), 3.6 mg kg^{-1} Q3W ($n=14$) and 4.8 mg kg^{-1} Q3W ($n=3$); 1.2 mg kg^{-1} weekly (Q1W) ($n=3$), 1.6 mg kg^{-1} Q1W ($n=2$), 2.0 mg kg^{-1} Q1W ($n=1$), 2.4 mg kg^{-1} Q1W ($n=13$) and 2.9 mg kg^{-1} Q1W ($n=2$).

Data processing for neural network. From the entire dataset, an 80% to 20% split of patients was made to divide them into the training ($n=532$) and testing ($n=133$) subsets. Data normalization was performed by taking the complete time course from the training set, and the normalization factors are computed to ensure each variable (that is, data columns including the time-after-dose, absolute time since the first dose, PK, platelet count, dosage) is normalized to have mean of 0 and standard deviation of 1; subsequently, the same scaling factors are directly applied to transform the testing set without any additional adjustments.

To combat overfitting, data augmentation on the set of training patients was performed as follows: we take the union of the following datasets, where in each case the notation input → output denotes that the network is fed input as data into the network and asked to predict the output as the prediction target:

- Complete time course: for training patient i,

$$\{\mathrm{PK}^i(t),\ \mathrm{Dosing}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty} \rightarrow \{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty}$$

- Observation data up to day 21: for training patient i,

$$\{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<21},\ \{\mathrm{Dosing}^i(t)\}_{0\le t<\infty} \rightarrow \{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty}$$

- Observation data up to day 35: for training patient i,

$$\{\mathrm{PK}^i(t),\ PD^i(t)\}_{0\le t<35},\ \{\mathrm{Dosing}^i(t)\}_{0\le t<\infty} \rightarrow \{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty}$$

- Observation data up to day 42: for training patient i,

$$\{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<42},\ \{\mathrm{Dosing}^i(t)\}_{0\le t<\infty} \rightarrow \{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty}$$

- Observation data up to day 63: for training patient i,

$$\{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<63},\ \{\mathrm{Dosing}^i(t)\}_{0\le t<\infty} \rightarrow \{\mathrm{PK}^i(t),\ \mathrm{PD}^i(t)\}_{0\le t<\infty}$$

After performing the above described data augmentation by cutting the data at different observation times, we have $532\times5=2{,}660$ set of augmented patient

records. The aim of the data augmentation process is to enrich the training dataset so as to force the neural network to achieve to goal of enable predictions in future time based having only the early observation data as well as the full dosing record.

Neural-PK model and training. The neural-PK model was constructed to reproduce the T-DM1 concentration time course predicted by the pop-PK model[18]. The schematic diagram of the model is shown in the upper part of Fig. 1a; further details on the model are shown in Extended Data Fig. 1a. The input data entering the network consists of a variable number of rows (denoted by n_1 in Extended Data Fig. 1a) and five columns, which are (1) time-after-dose, (2) time, (3) PK, (4) platelet count and (5) amount dosed. In particular, for our neural-PK model, the platelet count is not used as part of the input under the assumption that PK is not influenced by PD. Hence, only the columns 1, 2, 3 and 5 are fed into the PK encoder of Fig. 1a, which at its core contains a gated recurrent unit (GRU)[31] with a state size of 20, which finally outputs a four-dimensional vector after having gone through the sequence of data rows; for further details please refer to the Supplementary Software.

After having encoded the PK data, the model simulation is performed by a neural network of recurrent architecture that performs the forward Euler time steps[7,32] (further details are shown in Extended Data Fig. 1b). Dosing data enters explicitly in this recurrent network via the network port Dose, as shown in Fig. 1a. The PK vector field net aims to approximate the vector field of the PK model (its schematic diagram is shown in Fig. 1b). The layer $\times\Delta t$ performs element-wise multiplication by the step size (1/4 day in the current implementation) corresponding to the discretized time rate of change of the PK system. Finally, the ReLU[33] ensures that the updated state remains non-negative. The model construction was implemented in Wolfram Mathematica[34] and is provided in the Supplementary Software.

The trainable weights of the PK encoder and PK vector field net submodules are iteratively refined through a backpropagation algorithm[33] to minimize the L_2 loss function between the observed data and model outputs. As previously mentioned, we used an 80% to 20% split of the total patient data. Data augmentation was performed as described in the previous section. Note that once selected, the test patients are never used in the training phase but kept aside for the final evaluation of the network. The ADAM optimizer[33] was used to train the neural network for a total of 2,000 epochs. The trained network is then evaluated on the $n = 133$ test patients, and comparison to the ground truth showed good performance, with an R^2 of 0.98, a correlation coefficient of 0.99 and an r.m.s.e. of 2.67.

Pop-PK/PD model and prediction. We used a pop-PK/PD model with structure as described in refs. [17,18]. In particular, the model consists of a two-compartmental PK model (with two state variables and four parameters) and the platelet dynamics is described using six state variables and ten parameters that are variable between patients (that is, where inter-individual variability is allowed)[5]. The model was built sequentially[5,20], with PK parameters first estimated and subsequently followed by the PD parameters using the first-order conditional estimation method in NONMEM 7.3.0 (ref. [35]).

For the pop-PK/PD approach, we process data in the following manner: (1) for patients in the training set, we leave the whole observation and dosing data intact; (2) for patients in the test set, we keep all the dosing data but only retain PK and PD observation data within the initial window $t < t_{obs}$ (that is, setting DV to empty and MDV = 1 in the NONMEM[5,35] dataset). From this input data, NONMEM[35] was used to carry out estimation of the model parameters both at the population level and the individual level. From the result of the estimation, NONMEM produces the individual predictions (IPRED) for platelet dynamics among its outputs. We make predictions for the unseen portion (that is, $t \geq t_{obs}$) for patients in the test set and compare them with the data.

Neural-PD network and training. Having first trained the neural-PK net, we subsequently froze the weights of PK encoder and PK vector field net from any further training. The input PK/PD data (of dimension $n_1 \times 5$) is then selected for these four columns to enter the PD and PD initial condition encoders (see Fig. 1a): (1) time-after-dose, (2) time, (3) PK and (4) platelet count; that is, the dosing column is dropped. The implicit assumption being made is that it is the drug concentration that drives the pharmacodynamic effect, but dosing does not directly mediate the pharmacodynamic effect; this is a standard assumption made in PK/PD modelling. The PD encoder at its core contains a GRU[31] with a state size of 50 and outputs a ten-dimensional vector. Finally, using the outputs of the encoders the model simulation is performed using a neural network of recurrent architecture that performs the forward Euler time steps, which simulates both the PK and PD components of the model using those encoded vectors, driven by the dosing coming in via the dose network port as shown in Fig. 1. In our model, we represent the PD (platelet) dynamics using a four-dimensional ODE state vector. The underlying assumption that PK drives PD (but not vice versa) is encoded into the network architecture of Fig. 1c. For the PD dynamics, the corresponding four-dimensional ODE state vector needs to be initialized before the first dose is given; this is in contrast to the PK part of the model, where the drug concentrations are identically zero before dosing. The combined network (see Fig. 1a) takes the PD Initial State Estimate as an input, the first platelet count observed after the start of treatment ($t \geq 0$). However, simply estimating the platelet count before treatment by the first observed platelet data is inaccurate as some patients had a marked platelet drop right after the start of treatment; hence we use the PD initial condition encoder to help improve the estimation the initial platelet count. This was achieved by estimating a scalar from the early observed data and then replicating it to the four-dimensional platelet state vector. The need to estimate the initial platelet value from the time course data is similar to that of the pop-PK/PD myelosuppression models[17,19]. The architecture of the PD initial condition encoder consists of a bidirectional[36] GRU with a state size of 10 (see the Supplementary Software for further details on implementation). At each time step, the network outputs both the PK and PD (platelet count) predictions. In the sequential training approach, only the PD component is compared with the platelet data, based on the L_2 loss function (which computes the mean squared error). The ADAM optimizer[33] was used to train the neural-PK/PD network for a total of 3,000 epochs until the training loss no longer improves.

Model prediction benchmarking. We ensure that both the pop-PK/PD and neural-PK/PD are compared against the ground-truth platelet data in an identical manner on the test patients, with respect to both the R^2 and r.m.s.e.

Simulating dosing regimen. Although most patients in the training set were given doses at Q3W, a few were given doses at Q1W frequency. Nevertheless, we can use the neural-PK/PD model to simulate the scenario whereby all the patients were given T-DM1 at the dose 3.6 mg kg^{-1} Q3W as follows. First, the complete training patient data were fed into the network to obtain the encoded parameters outputs from network submodules PK encoder, PD encoder and PD initial condition encoder. We then feed the desired regular dosing of 3.6 mg kg^{-1} Q3W into the dose network port. The 5th, 50th and 95th quantiles of the neural-PK/PD simulated results from the training patients were then computed to generate the plots shown in Fig. 5a, which represent the between patient variability and are analogous to prediction intervals in a population simulation[2,4]. A similar procedure was used to simulate T-DM1 at the desired dosing regimens in the training patients to generate Fig. 5b and Fig. 5c.

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

Data availability

Source data are provided with this paper. Additional data that support the findings of this study are available from the corresponding author on reasonable request with the approval of Genentech. The raw data that underlies the findings of this study are not publicly available due to reasonable patient privacy concerns.

Code availability

The Wolfram Mathematica code for the neural-PK/PD model is available as Supplementary Software. It can be opened with the Wolfram Player Version 12.1.0 and above, which is freely downloadable from the link https://www.wolfram.com/player/.

Received: 22 September 2020; Accepted: 11 May 2021;
Published online: 21 June 2021

References

1. Meibohm, B. & Derendorf, H. Basic concepts of pharmacokinetic/pharmacodynamic (PK/PD) modelling. *Int. J. Clin. Pharmacol. Ther.* **35**, 401–413 (1997).
2. Upton, R. N. & Mould, D. R. Basic concepts in population modeling, simulation, and model-based drug development: part 3—introduction to pharmacodynamic modeling methods. *CPT Pharmacometrics Syst. Pharmacol.* **3**, 1–16 (2014).
3. Sheiner, L. B. & Ludden, T. M. Population pharmacokinetics/dynamics. *Annu. Rev. Pharmacol. Toxicol.* **32**, 185–209 (1992).
4. Bonate, P. L. *Pharmacokinetic–Pharmacodynamic Modeling and Simulation* 20 (Springer, 2011).
5. Owen, J. S. & Fiedler-Kelly, J. *Introduction to Population Pharmacokinetic/Pharmacodynamic Analysis with Nonlinear Mixed Effects Models* (John Wiley & Sons, 2014).
6. Keizer, R. J. et al. Model-informed precision dosing at the bedside: scientific challenges and opportunities. *CPT Pharmacometrics Syst. Pharmacol.* **7**, 785–787 (2018).
7. Chen, R. T., Rubanova, Y., Bettencourt, J. & Duvenaud, D. K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems 31* 6571–6583 (NeurIPS, 2018)..
8. Alber, M. et al. Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences. *NPJ Digit. Med.* **2**, 1–11 (2019).

9. Rackauckas, C. et al. Universal differential equations for scientific machine learning. Preprint at https://arxiv.org/abs/2001.04385 (2020).
10. Rubanova, Y., Chen, R. T. & Duvenaud, D. K., 2019. Latent ordinary differential equations for irregularly-sampled time series. In *Advances in Neural Information Processing Systems 32* 5320–5330 (NeurIPS, 2019).
11. Paoletti, M. E., Haut, J. M., Plaza, J. & Plaza, A. Neural ordinary differential equations for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **58**, 1718–1734 (2019).
12. Long, Z., Lu, Y. & Dong, B. PDE-Net 2.0: learning PDEs from data with a numeric-symbolic hybrid deep network. *J. Comput. Phys.* **399**, 108925 (2019).
13. Qin, T., Wu, K. & Xiu, D. Data driven governing equations approximation using deep neural networks. *J. Comput. Phys.* **395**, 620–635 (2019).
14. Zhang, T. et al. ANODEV2: a coupled neural ODE framework. In *Advances in Neural Information Processing Systems 32* 5151–5161 (NeurIPS, 2019).
15. De Brouwer, E., Simm, J., Arany, A. & Moreau, Y. GRU-ODE-Bayes: continuous modeling of sporadically-observed time series. In *Advances in Neural Information Processing Systems 33* 7379–7390 (NeurIPS, 2019).
16. Kidger, P., Morrill, J., Foster, J. & Lyons, T. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems* (NeurIPS, 2020).
17. Bender, C. B et al. An integrated PKPD model of trastuzumab emtansine (T-DM1)–induced thrombocytopenia and hepatotoxicity in patients with HER2–positive metastatic breast cancer. Preprint at https://www.page-meeting.org/pdf_assets/6110-PAGE%202016_bbender_FINAL.pdf (2020).
18. Bender, C. B. *Pharmacometric Models for Antibody Drug Conjugates and Taxanes in HER2+ and HER2– Breast Cancer*. Doctoral Dissertation, Acta Universitatis Upsaliensis (2016).
19. Friberg, L. E., Henningsson, A., Maas, H., Nguyen, L. & Karlsson, M. O. Model of chemotherapy-induced myelosuppression with parameter consistency across drugs. *J. Clin. Oncol.* **20**, 4713–4721 (2002).
20. Zhang, L., Beal, S. L. & Sheiner, L. B. Simultaneous vs. sequential analysis for population PK/PD data I: best-case performance. *J. Pharmacokinet. Pharmacodyn.* **30**, 387–404 (2003).
21. Lu, D. et al. Population pharmacokinetics of trastuzumab emtansine (T-DM1), a HER2-targeted antibody–drug conjugate, in patients with HER2-positive metastatic breast cancer: clinical implications of the effect of covariates. *Cancer Chemother. Pharmacol.* **74**, 399–410 (2014).
22. Chen, S. C. et al. Population pharmacokinetics and exposure–response of trastuzumab emtansine in advanced breast cancer previously treated with ≥2 HER2-targeted regimens. *Br. J. Clin. Pharmacol.* **83**, 2767–2777 (2017).
23. Fogel, A. L. & Kvedar, J. C. Artificial intelligence powers digital medicine. *NPJ Digit. Med.* **1**, 1–4 (2018).
24. Kelly, C. J., Karthikesalingam, A., Suleyman, M., Corrado, G. & King, D. Key challenges for delivering clinical impact with artificial intelligence. *BMC Med.* **17**, 195 (2019).
25. Nagendran, M. et al. Artificial intelligence versus clinicians: systematic review of design, reporting standards, and claims of deep learning studies. *BMJ* **368**, m689 (2020).
26. Chen, H., Engkvist, O., Wang, Y., Olivecrona, M. & Blaschke, T. The rise of deep learning in drug discovery. *Drug Discovery Today* **23**, 1241–1250 (2018).
27. Koch, G. et al. Pharmacometrics and machine learning partner to advance clinical data analysis. *Clin. Pharmacol. Ther.* **107**, 926–933 (2020).
28. Corrigan, B. W. Artificial intelligence and machine learning: will clinical pharmacologists be needed in the next decade? The John Henry question. *Clin. Pharmacol. Ther.* **107**, 697–699 (2020).
29. James, G., Witten, D., Hastie, T. & Tibshirani, R. *An Introduction to Statistical Learning* 112 (Springer, 2013).
30. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems* (O'Reilly Media, 2019).
31. Cho, K. et al. Learning phrase representations using RNN encoder–decoder for statistical machine translation. Preprint at https://arxiv.org/abs/1406.1078078 (2014).
32. Iserles, A. *A First Course in the Numerical Analysis of Differential Equations* (Cambridge Univ. Press, 2009).
33. Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016).
34. *Mathematica* v.12.1 (Wolfram Research, 2020); https://www.wolfram.com/mathematica
35. Beal, S. L., Sheiner, L. B., Boeckmann, A. J. & Bauer, R. J. *NONMEM 7.4.3 Users Guides (1989 –2018)* (ICON Development Solutions, 2018).
36. Schuster, M. & Paliwal, K. K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **45**, 2673–2681 (1997).
37. Klambauer, G., Unterthiner, T., Mayr, A. & Hochreiter, S. Self-normalizing neural networks. In *Advances in Neural Information Processing Systems* Vol. 30, 971–980 (NIPS, 2017).

Acknowledgements

We would like to thank D. Lu, L. Brooks, G. Liu, K. Liu, K. Deng and A. Joshi for their discussions and helping to make this work possible.

Author contributions

J.L. conceived the study. J.L. and J.Y.J. designed the study. J.L., B.B., J.Y.J. and Y.G. performed data analysis. J.L., B.B., J.Y.J. and Y.G. wrote the manuscript.

Competing interests

J.L., B.B., J.Y.J. are employees of Genentech and own stock in Roche. The work was performed when Y.G. served as a consultant to Genentech. The remaining authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-021-00357-4.

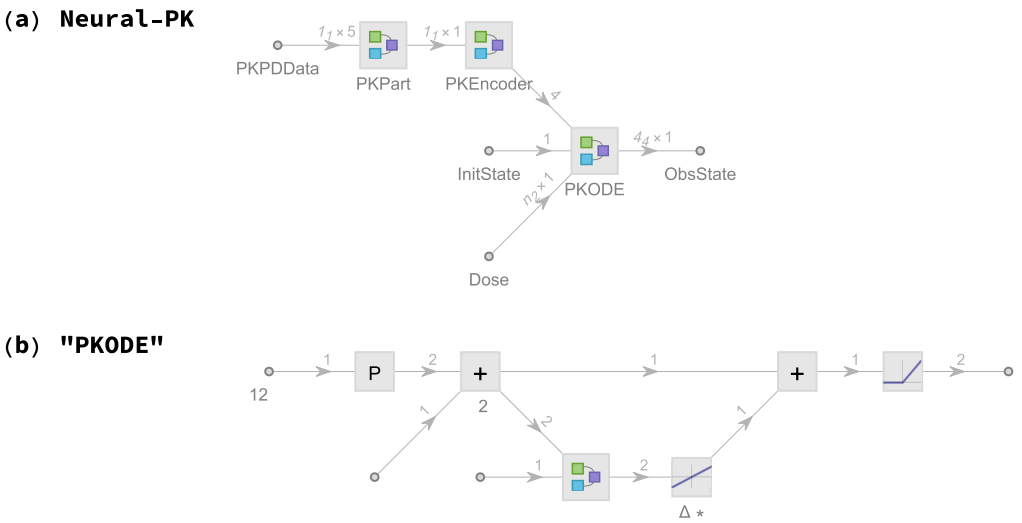
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-021-00357-4.

Correspondence and requests for materials should be addressed to J.L. or J.Y.J.

Peer review information *Nature Machine Intelligence* thanks Jorrit Enserink, Jing Tang and Giovanni Di Veroli for their contribution to the peer review of this work.

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

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Extended Data Fig. 1 | Supplementary Figure 1. a, The neural-PK schematic diagram. **b**, Detailed views into the 'PKODE' sub-module, which shows an implementation of the forward Euler step that explicitly incorporates the dosing data via the network port 'Dose'. The network shown in (**b**) is a recurrent network, which unfolds in time whereby successive 'NewPKState' is fed into 'PrevPK'. The 'Δt*' is a layer that multiplies the output of 'PKVF' by the constant step size Δt. The numbers in grey denote the dimension of the arrays involved in the computational graph.

nature research

Corresponding author(s): James Lu

Last updated by author(s): Apr 27, 2021

Reporting Summary

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

Statistics

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

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

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	No software specific to this study was used for data collection.
Data analysis	Wolfram Mathematica Version 12.1.0

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

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

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

Source data for Figures 2 to 5 are provided with the paper. Additional data that support the findings of this study are available from the corresponding author JL upon reasonable request with the approval of Genentech. The raw data that underlies the findings of this study are not publicly available due to reasonable patient privacy concerns.

Field-specific reporting

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

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

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

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

Sample size	This is a retrospective modeling study that uses previously compiled clinical data.
Data exclusions	Only 1 patient without any treatment data available was excluded in the current study.
Replication	Not applicable in current study
Randomization	Patients were randomly assigned to the training and test sets, without using any stratification factor.
Blinding	Not applicable in current study

Reporting for specific materials, systems and methods

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

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☐	☒ Human research participants
☒	☐ Clinical data
☒	☐ Dual use research of concern

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Human research participants

Policy information about studies involving human research participants

Population characteristics	The patients are 99% female with a median age of 53 years.
Recruitment	Not applicable in the current study.
Ethics oversight	All study designs were approved by independent ethics committees and conducted in accordance with the Declaration of Helsinki.

Note that full information on the approval of the study protocol must also be provided in the manuscript.