Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2983

A machine learning workflow to accelerate the design of *in vitro* release tests from liposomes

Daniel Yanes, [a] Vasiliki Paraskevopoulou,[b] Heather Mead,[b] James Mann, [b] Magnus Röding,[cd] Maryam Parhizkar, [e] Cameron Alexander, [a] Jamie Twycross*[f] and Mischa Zelzer *[a]

Liposomes are amongst the most promising and versatile nanomedicine products employed in recent years. *In vitro* release (IVR) tests are critical during development of new liposome-based products. The drug release characteristics of a formulation are affected by multiple factors related to the formulation itself and the IVR method used. While the effect of some of these parameters has been explored, their relative importance and contribution to the final drug release profile are not sufficiently understood to enable rational design choices. This prolongs the development and approval of new medicines. In this study, a machine learning workflow is developed which can be used to better understand patterns in liposome formulation properties, IVR methods, and the resulting drug release characteristics. A comprehensive database of liposome release profiles, including formulation properties, IVR method parameters, and drug release profiles is compiled from academic publications. A classification model is developed to predict the release profile type (kinetic class), with a significant increase in the balanced accuracy test score compared to a random baseline. The resulting machine learning approach enhances understanding of the complex liposome drug release dynamics and provides a predictive tool to accelerate the design of liposome IVR tests.

Received 19th March 2025
Accepted 26th August 2025

DOI: 10.1039/d5dd00112a

rsc.li/digitaldiscovery

Introduction

Liposomes are complex pharmaceutical products that, over recent years, have increasingly been used to overcome issues associated with poor aqueous drug solubility, toxicity, and lack of targeted delivery after drug administration.[1] During product development, the drug release behaviour of liposomes is a critical performance test that is performed to provide indications of safety, quality, and efficacy of the product. The drug release behaviour is assessed *via in vitro* release (IVR) tests.[2] The liposome formulation development process typically relies on a series of experimental tests to identify suitable formulation and/or IVR parameters to modulate the release profile. This experimental characterisation can be time-consuming, as there are no standard protocols to follow which causes delays in bringing liposome formulations to the market.[3]

Drug release from liposomes is dictated by multiple factors. The drug release profile depends on (i) critical material attributes (CMAs) such as drug and excipient properties, (ii) critical quality attributes (CQAs) of the formulation such as particle size, zeta potential, and drug loading,[4] and (iii) IVR test method parameters such as release medium temperature, medium pH, stirring speed, and release apparatus.[5,6] The characteristics of IVR test methods currently used to assess drug release from nanomedicines have been extensively reviewed before,[7–9] and the best fitting kinetic models were assessed for given drug release profiles of non-conventional dosage forms.[10,11] Nonetheless, the interplay between the parameters determining the profile is not well understood. To date, no attempt has been made to quantitatively link formulation characteristics, IVR test method parameters and the drug release profile of liposome formulations.

Machine learning (ML) has been used in different fields such as sustainable chemical reaction design,[12] nanomaterials characterisation,[13] and materials science[14] to accelerate processes and gain deeper insights into datasets and inform experiments. In pharmaceutical sciences, ML is now commonly used to establish connections between formulation and process parameters of drug delivery modalities to predict CQAs[15,16] and biological performance.[17] In addition, ML methods have been

[a]*School of Pharmacy, University of Nottingham, University Park Campus, Nottingham, NG7 2RD, UK. E-mail: mischa.zelzer@nottingham.ac.uk*

[b]*Global Product Development, Pharmaceutical Technology & Development, Operations, AstraZeneca, Macclesfield, SK10 2NA, UK*

[c]*Sustainable Innovation & Transformational Excellence, Pharmaceutical Technology & Development, Operations, AstraZeneca, Gothenburg, 43183 Mölndal, Sweden*

[d]*Department of Mathematical Sciences, Chalmers University of Technology, University of Gothenburg, 41296 Göteborg, Sweden*

[e]*School of Pharmacy, University College London, 29-39 Brunswick Square, London, WC1N 1AX, UK*

[f]*School of Computer Science, University of Nottingham, Jubilee Campus, Wollaton Road, Nottingham, NG8 1BB, UK. E-mail: jamie.twycross@nottingham.ac.uk*

used to predict drug release from various formulations, including 3D printed tablets,[18] long-acting polymeric injectables,[19] and oral formulations with polysaccharide coatings.[20] For lipid-based formulations, ML was applied to predict characteristics such as particle size,[21–23] liposome formation,[22] polydispersity index,[21] drug loading,[23] and encapsulation efficiency.[23] Prediction of drug release profiles from liposome formulations based on formulation and IVR characteristics using ML approaches has not yet been reported.

Here, we have curated a database from the existing literature on drug release from liposome formulations. Further, we have developed an ML workflow that explores and links drug release profiles from these systems with the formulation and IVR method characteristics. We identify which of the included factors are important for predicting drug release from liposomes. We furthermore establish a benchmark classification score for predicting types of drug release behaviour. The ML workflow presented here provides a deeper understanding of the complex relationship between nanomedicine CQAs, drug type, IVR method parameters, and their connections to kinetic drug release parameters.

Methods

Database construction and data acquisition

Literature data from liposome formulations were manually acquired and curated using three search methods to include data from specific FDA-approved liposome products and other, non-commercial formulations (SI, Section 2.1). The resulting articles were then sifted using a set of inclusion and exclusion criteria (SI, Section 2.2) to ensure that quantitative drug release data could be extracted and that the formulation and IVR method parameters were sufficiently detailed for entry in our database. Drug release plots were digitised from papers using *WebPlotDigitizer* (v4.6).[24] For articles that met the criteria, information relating to the search terms, article, compound used, formulation preparation, characterisation, composition, instrumental details, IVR testing conditions (apparatus, release media composition and conditions, specific methodological details such as volume of drug added) were recorded in a set of ten relational tables. The database schema was designed and implemented using *SQLite* (v3.43.1), integrated within a Python (v3.12.1) environment. Complete details of the database tables, including their primary and foreign key relationships, are reported in the associated GitHub repository (SI, Section 2.3).

Dataset construction

To extract information relating to formulation characteristics, IVR testing conditions and apparatus, lipid properties, and compound type into a dataset for data analysis, relevant queries were input to the SQL database (SI, Section 2.4). This resulted in a dataset with 271 observations, each with an associated digitised drug release profile. To ensure that the extracted plots were of sufficient quality for kinetic model fitting, a quality appraisal process was conducted on each of the 271 extracted drug release profiles (SI, Section 2.5). Subsequently, the plots that passed the quality appraisal process were subjected to automated batch parameter estimation using an in-house developed Python tool.

Batch parameter estimation and clustering of drug release profiles

Common kinetic models describing liposome drug release were fitted to each extracted drug release profile using least squares fitting (SI, Section 2.6). To improve the likelihood of finding a global minimum over a local one during parameter optimisation, 100 initial parameter sets for each model were generated using Latin hypercube sampling, ensuring a comprehensive exploration of the parameter space. Each initial parameter set were used to estimate model parameters for each drug release profile. The best fits were selected with respect to the mean absolute error (MAE) (eqn (S9)). Detailed information on parameter bounds, optimisation method, and implementation is provided in the SI, Table S2. Subsequently, model fitting was evaluated using a suite of metrics (SI, Section 2.7). A threshold dissolution similarity factor (f_2) (SI, Section 2.7.4) score of greater than 50 was selected based on FDA guidelines.

Drug release profiles were clustered using principal component analysis (PCA) and *k*-means clustering (KMC). Weibull model parameters with f_2 scores below 50 and those displaying two release mechanisms (biphasic), which were not described sufficiently well by the Weibull model were excluded. To enable clustering of drug release profiles based on similar shapes, a simulated drug release dataset was generated using fitted Weibull parameters. As the original database contained release profiles with varying time scales, including seconds, minutes, hours, and days, the lack of standardisation of the time range of the investigation, would mean that a profile initially measured in seconds but simulated over 24 hours would automatically be classified as fast. Since hours was the most common unit, only profiles originally measured in hours were selected. The fitted Weibull parameters of the resultant profiles were used to simulate drug release data over 0–24 hours, resulting in a dataset with 500 time points as columns and rows representing individual simulated profiles. The dataset was standardised and reduced to a low-dimensional representation *via* PCA. The first two PC scores were clustered using KMC with the number of clusters (k) ranging from 2 to 8. The clustering quality was evaluated by assessing cohesion and separation for each k (SI, Section 2.8).

Data preprocessing

The classification model's target output was defined by the resulting assignment of each profile subjected to PCA and KMC, as each profile in a cluster resembled similar shapes this defined the type of release (kinetic class). The number of clusters (k) was chosen as 3, representing slow, medium, and fast kinetics classes, which defined the target output. To train the ML classifiers, feature inputs were added to the target output, creating a suitable dataset. The unique drug release profile identification number (*IVR_ID*) was used to merge back-end data, including information relating lipid properties, drug

type, formulation characteristics, and IVR testing conditions to the target output. Data preprocessing steps included removing features with many missing values, eliminating rows with missing data, deleting duplicate rows, and standardising. Details of feature inputs, data types, encoding, and implementation are provided (SI, Section 2.9).

Machine learning modelling and evaluation

To predict the kinetic class given feature inputs (SI, Table S3), a range of ML classifier models were evaluated. The models selected were DecisionTreeClassifier (DTC), SupportVectorClassifier (SVC), GaussianNaiveBayes (GNB), KNeighborsClassifier (KNC), LogisticRegression (LR), RandomForestClassifier (RFC), and ExtremegradientboostingClassifier (XGB). All models were implemented through the *Sci-kit learn* package[25] (v1.4.0), except for XGB, which used the *xgboost* package[26] (v2.023). The default hyperparameters were selected for each model.

To evaluate the performance of each model, the target output (kinetic class) obtained using each model was compared to their true labels determined by PCA and KMC. The classification task was multi-class and imbalanced, therefore careful selection and interpretation of metrics compared to simpler binary tasks was required. Briefly, micro-averaged F_1 score, balanced accuracy, and Matthews correlation coefficient (MCC) (SI, eqn S17, S19 and 24) were used to ensure evaluation of global and class-specific performance. To determine whether the performance between classifiers were statistically significant, pairwise comparisons were conducted using the Wilcoxon Signed-Rank test across cross-validation folds. Additional details of the rationale, equations, and implementation of the classification metrics and statistical tests are provided (SI, Sections 2.10, 2.11.1).

Shapley Additive exPlanations (SHAP) analysis[27] was conducted on the XGB classifier trained on all available data. XGB classifier was chosen due to the ability to handle complex relationships and good predictive accuracy on small tabular datasets.[28] SHAP values were calculated for every feature and each instance to quantify input contributions. Beeswarm plots were produced using the *SHAP* library (v0.44.1).

The optimal subset of feature inputs to predict the kinetic class was determined for the XGB classifier using backward elimination with cross-validation. Briefly, the classifier is trained on all 9 feature inputs, and the importance of each feature using the gain feature importance attribute of the XGB classifier model is determined. The least important features are pruned from the current set of features, the procedure is recursively repeated on the pruned set until 1 feature remains. The optimum subset of features was determined as the one yielding the maximum 5-fold cross-validated test score with minimum variance.

The optimum subset of features was used to assess the performance of each classifier on unseen (test) data. The cleaned dataset ($n = 77$) was split into training and test sets using stratified 5-fold cross validation, to ensure a similar class distribution in each fold. Due to 5 not being a divisor of 77, the training and test sets consisted of 61–62 and 15–16 samples respectively. On each fold, the balanced accuracy, MCC, and F_1 score were calculated and the mean score and standard deviation across the 5 folds were calculated.

Permutation testing was conducted to evaluate whether the average cross-validated balanced accuracy score of the best performing classifier significantly exceeded that of a three-class random classifier. Target labels were permuted 1000 times while keeping all input features unchanged. For each permutation, the balanced accuracy was calculated. An empirical p-value was then calculated to estimate the probability of obtaining the observed average balanced accuracy score by chance. This calculation used the number of permutations with scores greater than or equal to the average cross-validated test score obtained by the best performing classifier (C), given the number of permutations (S) (eqn (1)). The choice of number of permutations was validated by assessing the empirical p-value as a function of the number of permutations (Fig. S9). Details of implementations and rationale are provided in the SI, Section 2.11.

$$p = \frac{(C+1)}{(S+1)} \tag{1}$$

Results and discussion

Assembling a dataset suitable for a multi-class classification model

To develop a multi-class classification model to accelerate the design of liposome IVR tests, it was necessary to first establish a data workflow. The snapshot in Fig. 1 highlights the workflow developed to achieve the main objective of the study: linking IVR testing conditions, lipid properties, drug type, and formulation characteristics to drug release profiles, to streamline choices made during liposome IVR testing.

The first step was to assemble a dataset suitable for the classification task of predicting release kinetics from liposomes. To achieve this, a structured query language (SQL) database was compiled using data extracted from 34 academic publications. The database included 141 distinct formulations, 22 active compounds, and 31 excipients (lipids), culminating in a total of 271 IVR tests. Each IVR test featured unique formulations with varying numbers and types of excipients, along with different characterisation data, compounds, and experimental IVR testing conditions. The SQL database provided a well-defined structure, making it advantageous for organising such complex and variable data. As high throughput techniques become increasingly prevalent,[29–31] our database, compared to storing tabular data in Excel, ensures the efficient, scalable, and secure management of large datasets generated across multiple formulations, testing conditions, and drug compounds.

To establish connections between the four distinct groups of variables (Fig. 1) stored in the database and drug release profiles, nine features were selected (SI, Table S3). By organising the variables into the groups outlined (Fig. 1) and using the database schema, this approach can be adapted to other drug

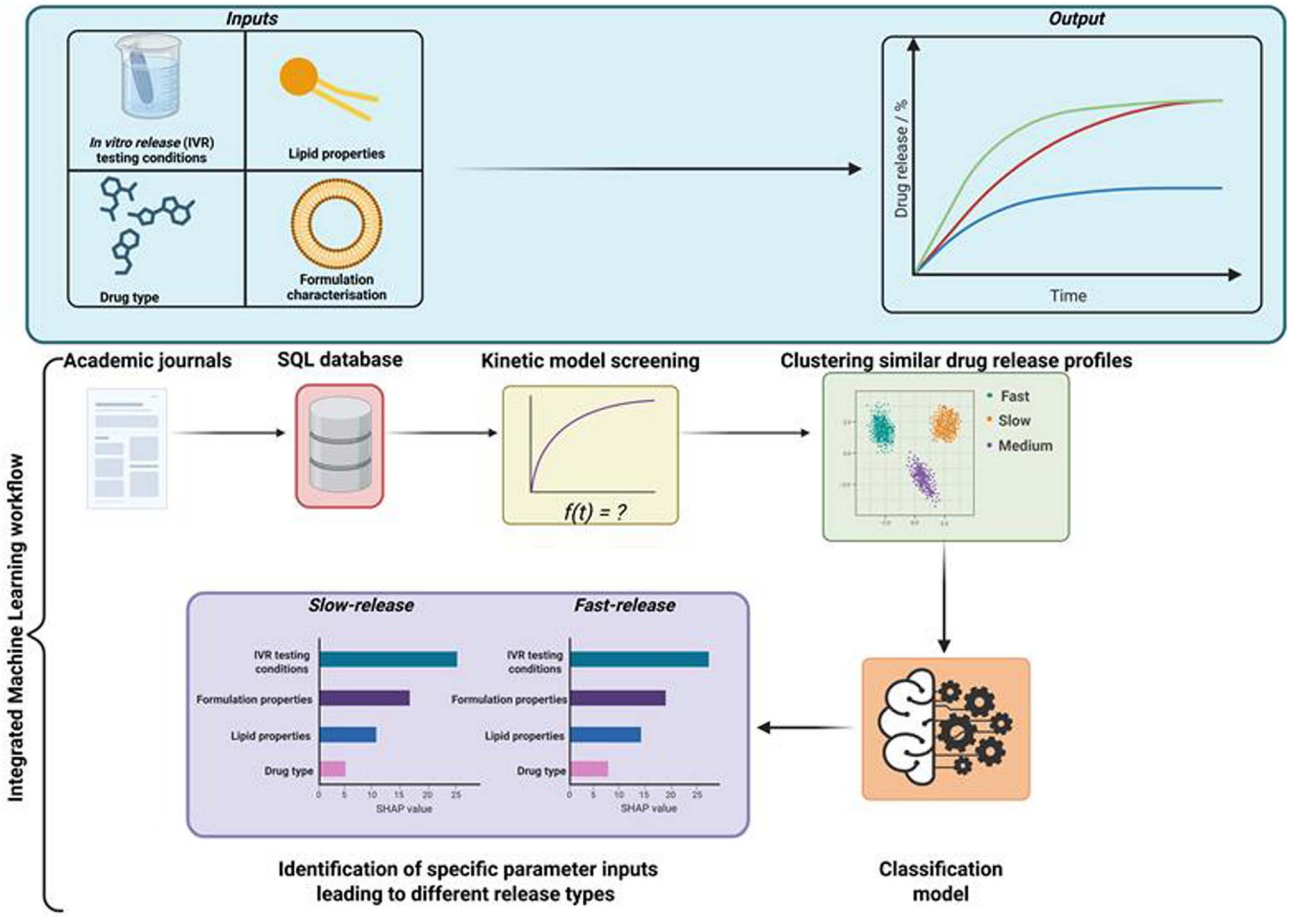

Fig. 1 Problem statement, integrated analysis and workflow summary. Traditional experiments yield insufficient data to understand complex multivariate processes such as drug release from liposomes. Storing historic data generated during formulation development in a centralised, standardised repository (Structured Query Language, SQL database) enables quantitative relationships between feature vectors (inputs) and drug release profiles (output) to be established and analysed. An in-house Python tool was developed for the batch parameter estimation of digitised drug release plots to obtain fitted kinetic model parameters (output) to define drug release profiles. To isolate specific feature inputs driving distinct types of drug release behaviour, the extracted release profiles were grouped into clusters which exhibited similar temporal patterns. From these clusters, various classifiers were trained to connect feature inputs to the type of release profile (slow, medium or fast). This enabled identification of key drivers of types of liposome release behaviour. A benchmark classification score was also established for the classification of release kinetics from liposomes *via* train-test splits using 5-fold stratified cross validation.

delivery modalities. For instance, the lipid properties group could be replaced with polymer properties to use the workflow for connecting drug release for nanoparticle, dendrimer or hydrogel systems.

Liposomes are formulated with a variable number of lipids, at different lipid ratios of different lipid types. Hence, for convenience, the lipid-based features were reduced to a single scalar value per formulation. Here, the molar-weighted excipient molecular weight (M_w) and weighted lipid phase transition temperature (T_p) were selected, as T_p links lipid structure and size to drug release behaviour.[32] Since no standardised methods exist for IVR testing of liposome products,[7] the impact of user-selected apparatus, and release medium conditions was also explored. This comprehensive feature selection aimed to identify the most important features for predicting liposome drug release.

To establish relationships between the assembled feature input vectors and the drug release profiles, we described drug release profiles by parameterising the entire release profile over time. This is an alternative approach to reducing release profiles to discrete points such as T_{20}, T_{50}, and T_{80} (times to reach 20%, 50%, and 80% drug release), commonly used in ML-based drug release prediction tasks.[18] To our knowledge, this represents a novel approach to drug release prediction which involves automated batch parameter estimation of drug release profiles, unsupervised learning to group similar profiles, and feature importance analysis to identify drivers of types of drug release behaviour. By fitting kinetic models to release profiles, this method captures the complete release kinetics, providing deeper insights into the drug release process and more flexibility for data analysis.

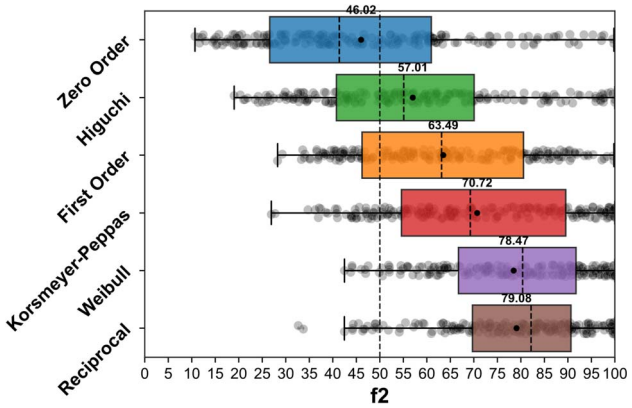

Fig. 2 f_2 score of each drug release profile (datapoint) fitted to six kinetic models to describe drug release. Models ordered based on increasing average f_2 score (bold).

Batch parameter estimation of drug release profiles

The aim of this part of work was to perform a statistical comparison of common kinetic models used to describe drug release behaviour. The goal is to identify a model that can correctly describe most of the extracted drug release profiles in the database. This yields a set of parameters that can be used for subsequent processing of the drug release profiles. Given the variable reporting quality of the drug release profiles in the literature, a quality appraisal of the 271 drug release profiles was conducted (SI, Section 2.5), resulting in keeping 209 profiles deemed suitable for model fitting.

Software tools for drug release curve fitting are often limited to using one model at a time, for example by using the Excel plugin DDSOLVER.[33] This approach is not suitable for assessing multiple models across numerous drug release profiles. To overcome this, a custom in-house Python tool was developed for batch-wise parameter estimation of drug release profile models. The 209 drug release profiles were subjected to screening using the tool, where six commonly employed kinetic models (SI, Section 2.6) that are used to describe drug release profiles were selected. These are denoted zero order, first order, Higuchi, Korsmeyer–Peppas, Weibull, and Reciprocal.

Based on values of mean relative root mean squared error (RRMSE) and Akaike information criterion (AIC) for each drug release profile, the Reciprocal model demonstrated the lowest

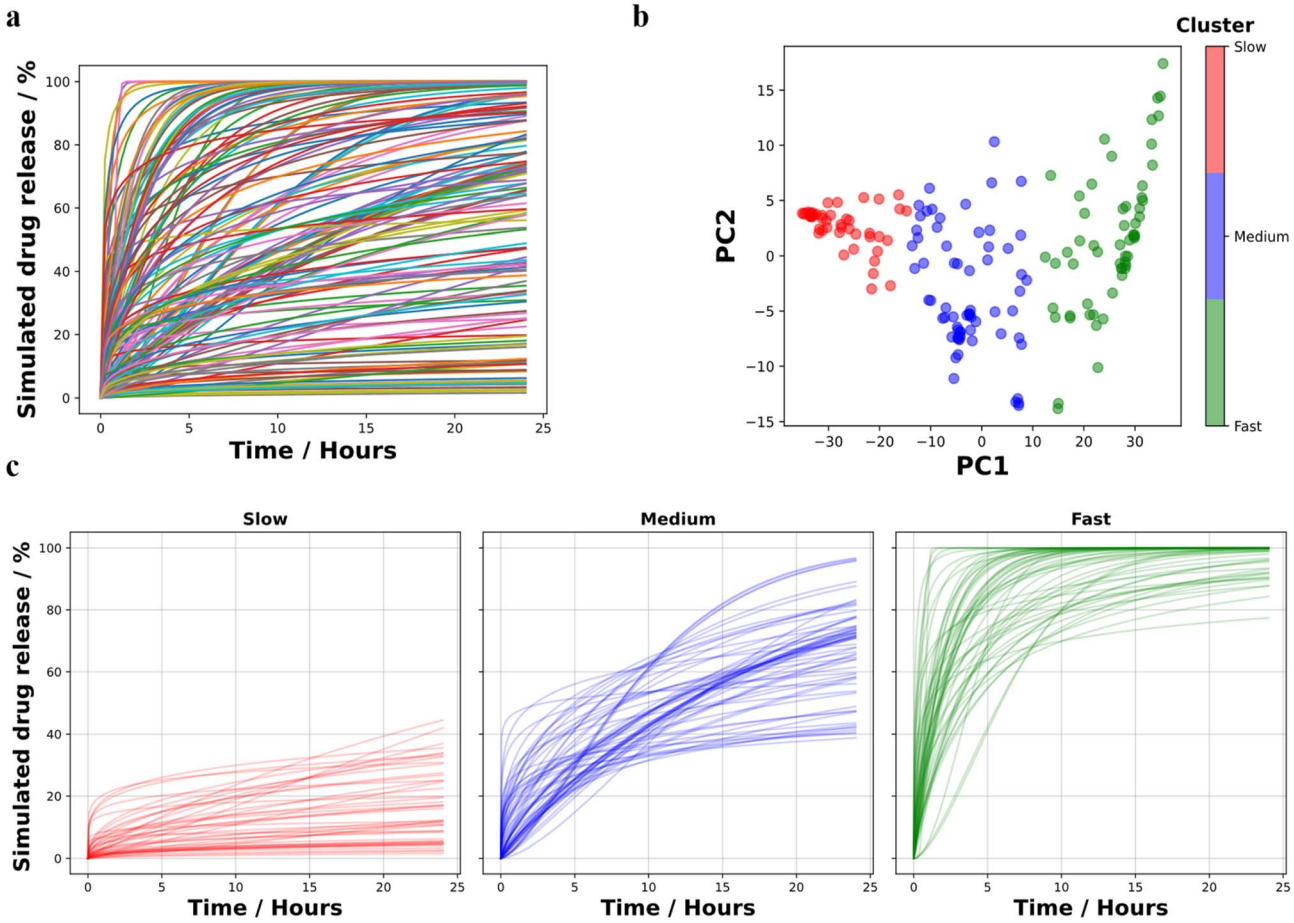

Fig. 3 Principal component analysis (PCA) followed by k-means clustering (KMC) PCA-KMC used to group similar extracted drug release profiles. (a) Simulated drug release profiles generated using fitted Weibull parameters to visualise dataset complexity and need for unsupervised clustering. (b) Transformation of simulated drug release profiles (Results section: batch parameter estimation and clustering of drug release profiles) into principal component (PC) scores, coloured by cluster assignment of k-means clustering (KMC) that minimises within sum of squares variation (WSS) and maximises silhouette coefficient. (c) Simulated Weibull drug release profiles assigned based on PCA-KMC ($k = 3$).

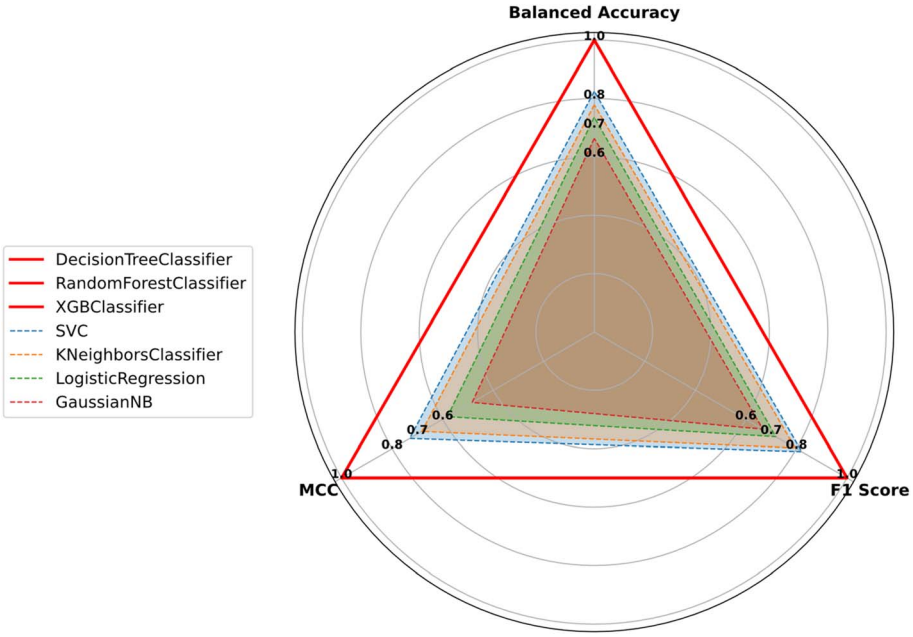

Fig. 4 Radar plot summarising performance of screened models, trained on all available data (n = 78). Table S7 shows the numeric data. RandomForestClassifier, DecisionTreeClassifier, and XGBClassifier had identical metrics (all three models are represented by the same red lines) and outperformed all other models, classifying all training samples correctly.

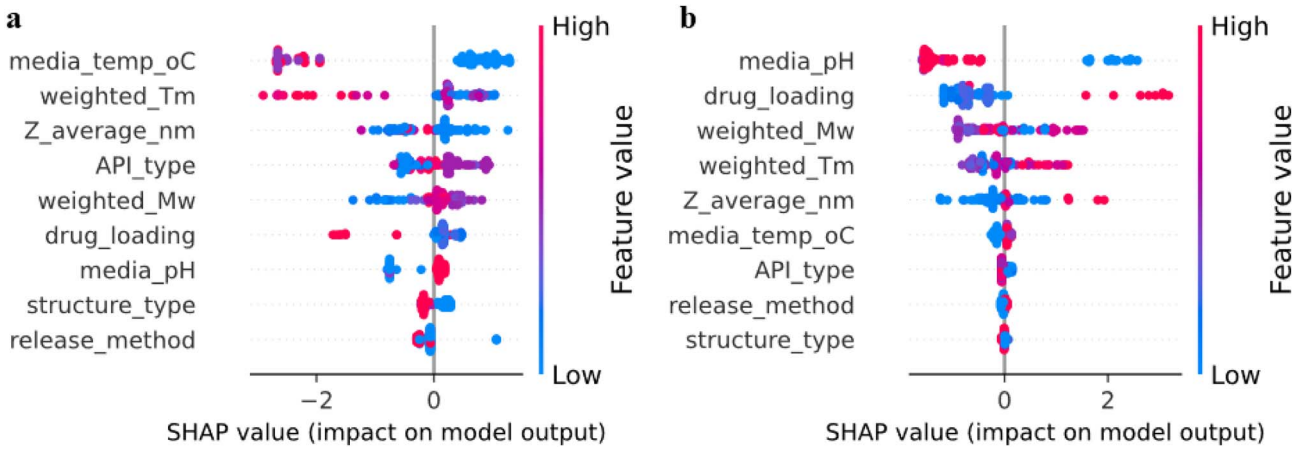

Fig. 5 Ranking of features contributing to prediction of slow (left) and fast (right) release kinetics using a XGBoost model trained on all available training data (n = 78). Features ranked by importance (top-bottom). Each instance was correctly predicted in training data, by interrogating the training process reveals drivers of slow and fast-release profiles. (a and b) SHAP beeswarm plot displaying individual SHAP values for a feature from each instance coloured by feature value. Features with higher SHAP values correspond to increasing probability of belonging to slow-release (a) and fast-release (b) kinetic classes.

average RRMSE (0.08) and AIC (30.20) (SI, Fig. S1), indicating a superior fit compared to other models. To further evaluate its performance, a Kolmogorov–Smirnov test was conducted to assess whether the absolute error distribution of the Reciprocal model was smaller than that of the Weibull model. The test yielded a p-value of 0.77, indicating no significant difference between the two models' error distributions (SI, Fig. S2 and Table S4). These findings align with previous studies comparing kinetic models for liposome systems, where both the Reciprocal and Weibull models provided acceptable fits. However, the Reciprocal model showed better fitting for thermosensitive liposomes, particularly those smaller than 100 nm.[11]

To define an accuracy threshold for parameters yielding sufficiently similar simulated and experimental drug release profiles, the similarity factor (f_2) was used with a cutoff value of 50, based on FDA guidelines, where higher values indicate greater similarity.[34] While the Reciprocal model achieved the highest average f_2 score, the Weibull model covered a greater proportion of profiles with f_2 greater than 50 (Fig. 2). Based on our need for greater data availability and the advantage of

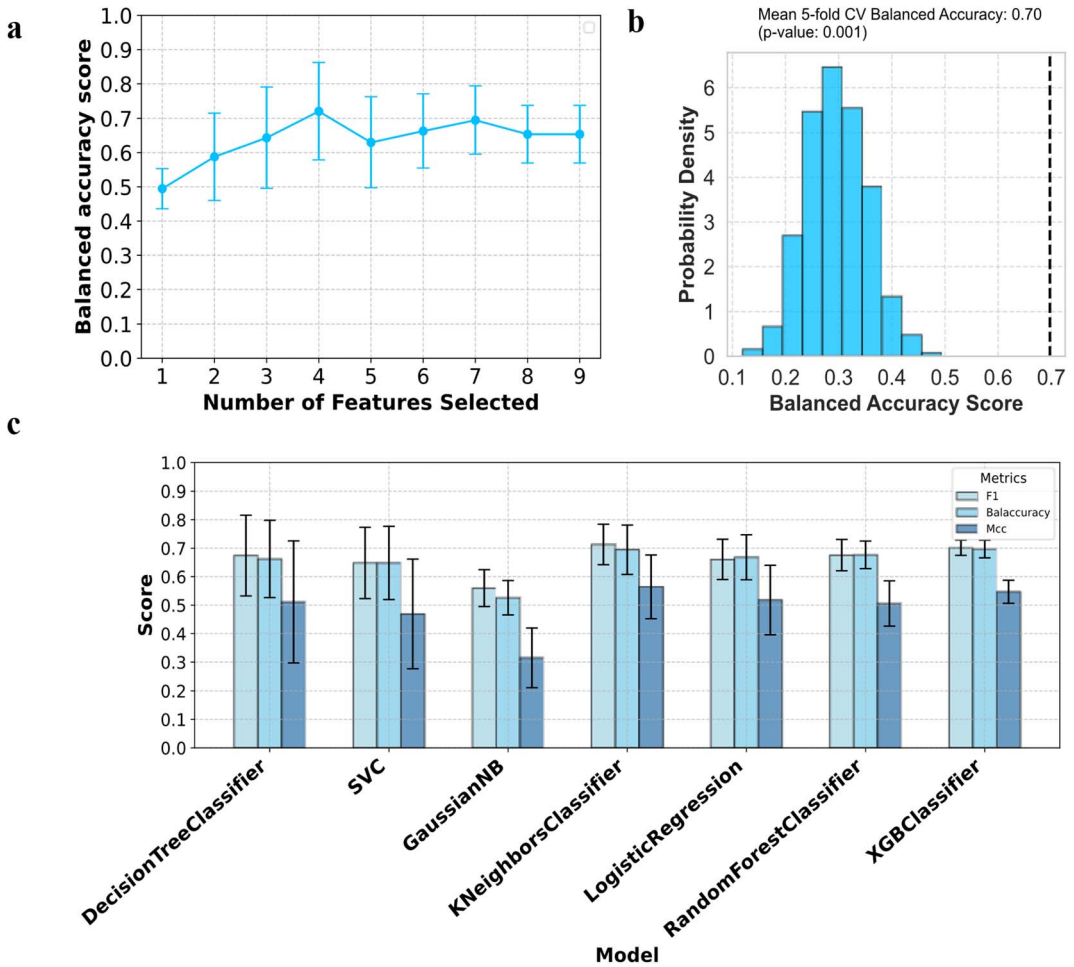

Fig. 6 Determining optimal number of feature inputs and performance on unseen data. (a) Backward elimination with cross validation to select number of features to maximise cross validated balanced accuracy score for XGB classifier. Features 8 and 9 represent release method and structure type, respectively. (b) Permutation test results for the XGB classifier trained on 7 most informative features. The histogram represents distribution of classifier accuracy on 1000 different permutations of the dataset, where features remain constant, but target output labels undergo different permutations. The distribution obtained represents the null hypothesis stating there is no dependency between features and labels. The black dotted line represents average cross validated balanced accuracy score on the original (unshuffled) dataset. A permutation-based p-value was calculated, representing how likely the classifier performance is obtained by chance, where the number represents the fraction of permutations where the balanced accuracy score is greater than the score obtained using the original dataset. (c) Bar plot summarising the performance of surveyed models which shows the XGB classifier model outperformed all other models across all evaluation metrics. Each model was trained using 5-fold stratified cross validation at a train : test (80 : 20) split using the 7 most informative features. Each model was evaluated using F_1 score, balanced accuracy, and Matthews correlation coefficient (MCC).

a more generalisable applicability of the chosen model to diverse datasets, the Weibull model was selected.

Plotting each drug release profile together with the fitted Weibull model (SI, Fig. S3 and S4) visually demonstrated that the Weibull model fits the data well in most cases. Examining profiles near the cutoff range (f_2 scores 45–55) shows that the Weibull model failed to describe biphasic release patterns (SI, Fig. S3). As a result, profiles with f_2 values of less than 50 and biphasic release were excluded, leaving 197 profiles where the Weibull model provided acceptable similarity to the original release profiles. With current software tools, this analysis would have required 1254 individual Excel files. Our Python tool enabled the analysis to be conducted in batch, making it suitable for large volumes of drug release data and thus more applicable to emerging trends in the field.

Grouping similar drug release profiles *via* unsupervised learning

The next step in developing a multi-class classification model to predict liposome release kinetics, was to group similar drug release profiles in an unsupervised, automated manner which would define the target output. To do so, the 197 drug release plots accurately described by the Weibull model were first selected. Then, to standardise the time scale of the investigation, only those originally done in units of hours were selected, leaving 169 profiles (Methods section: batch parameter estimation and clustering of drug release profiles).

Using the fitted Weibull parameters (α, β) of these profiles, a simulated drug release dataset was generated, capturing a variety of release behaviour (Fig. 3a). Principal component analysis (PCA) was used to transform the high-dimensional release profiles into a lower-dimensional space while retaining 95% of the variance in the first two PCs (SI, Fig. S5). The resulting PCA score plot represented each drug release profile as a point (Fig. 3b). Profiles represented by points closer together were considered to have similar underlying data structures, *i.e.*, similar drug release behaviours. To group the PC scores of the first two dimensions into a set of distinct clusters, where data points in each cluster would share common data structures, *k*-means clustering was used (Fig. 3b). A range of k values were explored to minimise the within sum of squares (WSS) variation and maximise the silhouette coefficient (s_i), representing cluster compactness and separation, respectively (SI, Fig. S6). The number of clusters was set to 3, reflecting a reasonably minimal WSS (SI, Fig. S7) and maximum average s_i across each cluster (SI, Fig. S8). Based on the *k*-means cluster assignment, each drug release plot was assigned into a respective cluster, which was used to simulate the drug release behaviour within a given cluster (Fig. 3c). This revealed that each cluster assignment contained distinct types of drug release behaviour. Specifically, the slow release was less than 50% release, medium was characterised by gradual release and fast was characterised as burst release. Overall, the PCA-KMC process facilitated the assignment of target outputs to develop a multi-class classification model *via* machine-informed classification of drug release profiles, eliminating the need for manual annotation of the 169 profiles. While this approach has proven successful for the present dataset, it must be noted that the generalisability of the approach has yet to be established and translation of the approach to different kinetic models would have to be verified.

Training scores and feature importance

To understand which features were important in predicting the type of release profile (kinetic class), a library of seven ML classifiers was screened on the cleaned training data ($n = 78$). This enabled us to assess whether classification models could approximate the relationship between feature vectors and types of drug release. The classification task faced was multi-class (slow, medium, and fast) and the class distribution (number of observations in each class) was 25, 34, and 19, respectively. Three evaluation metrics were selected to obtain a full assessment of classifier training performance (Results section: machine learning modelling and evaluation). The perfect training classifier performance of DecisionTreeClassifier (DTC), RandomForestClassifier (RFC), and XGBClassifier (XGB) showed the tree-based algorithms (DTC, RFC, and XGB) outperformed the simpler models (SVC, KNC, LR, GNB) (Fig. 4 and ESI Table S6). This suggests that the nature of the data was complex and non-linear as the training data did not fit the simpler models well, highlighting the benefit of using more complex ML models for this dataset.

Here, for the 9-feature XGB model, Shapley Additive exPlanations (SHAP) analysis was used to comprehend the model's decisions and provide insight into the contributions of each feature to the prediction.[35] SHAP values represent contributions of each feature to the model output,[27] *i.e.*, the kinetic class (slow, medium, and fast) in our case. The SHAP analysis will thus provide an indication of how much a feature contributes to the prediction output, *i.e.*, one of the drug release classes: slow, medium or fast. The magnitude of the SHAP value gives indication of the importance of a feature on making a prediction, were positive SHAP values show features pushing model output towards a prediction.

To identify whether higher or lower values of specific feature(s) increase or decrease the prediction of release kinetics, a bee-swarm plot was used to visualise SHAP values for slow and fast release predictions (Fig. 5a and b). Slow and fast release predictions were selected as extremities, but the same analysis could have been conducted for medium release. For example, lower media temperature and T_p made positive contributions to the slow-release class. Bearing in mind the lipid bilayer structure of liposomes, it follows that lowering the media temperature reduces the lipid bilayer membrane permeability, thus causing a reduction in drug efflux. The lower T_p driving slow release is linked to the formulation composition and can therefore be adjusted by the choice of lipid type and by varying the ratio of different phospholipids.[36]

Other noticeable positive contributions to slow-release classification were made by higher media pH values and higher excipients M_w. The compound in the slow-release class was predominantly doxorubicin (14/25) which has a pK_a of 8.2. In contrast to the fast release scenario, where low pH values are expected to lead to increased ionisation and hence increased solubility of doxorubicin, slow-release kinetics would be observed at higher media pH values where doxorubicin would remain non-ionised. Unlike media pH, which is well known to be of importance for drug release, the importance of M_w of excipients used in the formulations has not been reported. This could be a potential avenue for future exploration, for example selecting higher M_w excipients to make the formulation such as 1,2-distearoyl-*sn*-glycero-3-phosphoethanolamine-poly(ethylene glycol) (DSPE-PEG-10k).

Across slow and fast release, the lamellarity of the vesicle (structure type) and user choice of IVR apparatus had the least influence on the prediction (Fig. 5a and b) of kinetic class. This is expected, because the method developed is properly validated to separate free drug from the nanocarrier and hence, the user choice of apparatus used for the IVR test should not influence release kinetics. This is shown by a comparison of ultracentrifugation and continuous flow sampling methods for polymeric nanoparticles,[6] this suggests the same finding extends to liposome systems.

It is worth noting that categorical features such as release method, structure type, and API type were label encoded, removing the original meaning. It must be noted that the drawback of this method is that the selection of integers that are assigned to each categorical variable could influence the SHAP analysis and other subsequent analysis. This could be mitigated by one-hot encoding, which would preserve the original feature input by creating separate binary columns for each category

type, *e.g.*, doxorubicin: yes/no; however, the drawback of this approach is an increase in the dimensionality of the dataset. As we aimed to minimise the complexity of the dataset, we decided against implementing one-hot encoding in this work.

Overall, the SHAP analysis was used to identify and rank features impacting types of drug release from liposomes. It is clear, and hardly surprising to experimentalists, that IVR conditions (temperature, pH) play a key role in determining drug release kinetics. As IVR tests are typically done at physiological conditions (37 °C), the value of this analysis lies in the insight it provides on the effect of formulation properties. The choice of lipid, and the M_w of excipients are key drivers to obtain slow-release lipid based formulations for the drugs included in this study (SI, Table S5). It should be noted that the SHAP analysis provides insights for the compiled and cleaned dataset fitted to a XGB model, it remains to be seen if these insights can be generalised to all liposomes and different models. Additionally, a feature identified as being important for a prediction does not always equal a causal relationship.

Assessing classifier performance on unseen data: towards the prediction of drug release from liposomes

Perfect training performance (Fig. 4 and SI Table S6) indicates that the model has overfit by learning all the minute patterns in the training data, which suggests it may perform poorly on new, unseen data (generalisation).[37] In this case, the overfitting is likely due to the limited dataset size due to extensive data cleaning from 271 to 77 observations due to data heterogeneity.

To examine how well the XGB classifier model with default hyperparameters would perform on unseen data, stratified five-fold cross validation was used to assess the ability of the model to generalise and gauge the range of prediction error. To achieve this, the data was split 80 : 20 into training and testing sets respectively. Each of the 5-folds were stratified such that the class distribution in each set was as similar as possible.

Prior to this, it was necessary to determine which combination and number of input features led to the maximum cross validated balanced accuracy test score with lowest variance. To do so, backward feature elimination was used. This showed that a selection of 7 features was optimal for maximising balanced accuracy whilst minimising variance (Fig. 6a). Addition of features relating to the testing apparatus used and structure type decreased the mean balanced accuracy score, which aligns with the SHAP analysis (Fig. 5). Variations in trends of number of features *versus* accuracy such as those observed here have been reported previously for backward feature elimination.[38–40]

The *release_method* and *structure_type* features were identified as least influential, aligning with the SHAP analysis. These two parameters were subsequently excluded, resulting in 7 features that were used as inputs to screen the performance of the 7 classifiers. The screening showed that the XGB classifier model had the highest balanced accuracy score with the lowest variance (0.70 ± 0.03) (Fig. 6c, ESI Tables S7 and S8). To assess whether the XGB model significantly outperformed the other classifiers, across all metrics, a pairwise Wilcoxon Signed-Rank test was applied to the individual fold results per classifier (SI Tables S9 and S10). The analysis revealed no statistically significant differences in evaluation metrics between models, suggesting no single classifier consistently outperformed the others. Although there is no statistical difference in model performance, we suggest to use XGB for this dataset because of the XGB classifier shows the lowest variance, suggesting that it might result in a more reliable and stable performance.

The cross validated training and testing scores for the XGB model demonstrated a large difference, indicative of overfitting, which could be prevented by regularisation *via* tuning model hyperparameters. Nonetheless, the XGB model had the highest cross validated balanced accuracy score of 0.70 ± 0.03 ($n = 5$) with the lowest variance, reflecting 10–11 correct predictions out of 15 unseen examples. The classifier score obtained is reasonable considering that, as the number of classes to predict increases, it becomes harder to achieve higher accuracy scores.[41]

To assess whether the accuracy score on unseen data improved beyond random predictions of slow, medium or fast release kinetics, the classification labels were permuted 1000 times to remove dependencies between features and targets. The resulting permutation scores, representing the null distribution, showed an average balanced accuracy of approximately 0.33, as expected for a three-class classification task. The XGB model mean cross validated scores on the original data, indicated by the black dotted line in Fig. 6b, demonstrated significant improvement over the permuted data (p-value = 0.001). This suggests that the observed balanced accuracy is unlikely a result of chance, confirming dependencies between features and target outputs.[42]

These results establish the model's ability to outperform random guessing and sets a benchmark for classifying liposome release kinetics, which can make liposome IVR test design more rational. With the recently proposed DELIVER framework[43] which aims to reduce nanomedicine development timelines and clinical trial failures, this computational workflow could integrate IVR and formulation data generated during lead discovery and optimisation. The DELIVER framework has identified experimental testing conditions that lead to uncontrolled drug release as a key risk blocking nanomedicine development timelines. The present approach may have the potential to de-risk preclinical evaluation by providing greater understanding of the drug release data generated in a *post-hoc* analysis, to help identify formulation(s) and/or experimental testing conditions that may lead to uncontrolled drug release. In the future, it would be of benefit to have real-time risk quantification of conditions leading to uncontrolled release.

Conclusion

The use of ML models to explore complex multivariate processes such as drug release from liposomes can help identify key parameters driving the distinct types of release behaviour. This enables a ranking of experimental and/or formulation parameters on the prediction of drug release kinetics and thus could provide better means to modulate drug release profiles. In this work, a computational workflow is developed to gain

insight into the relationship between drug type, experimental IVR testing conditions, formulation, and lipid properties to drug release profiles.

By using SHAP analysis on a dataset of liposome drug release, the user choice of IVR testing conditions has greater influence on predicting release kinetics than drug type, lipid composition, and formulation characteristics. Furthermore, by using widely available input parameters, an XGB classifier model is identified as a stable model to predict the class of drug release profile (slow, medium or fast). Compared to a random baseline classifier, there is a significant improvement in balanced accuracy score from 0.33 to 0.70 $\pm$ 0.03, indicating that the model has learnt a real relationship between feature inputs and target outputs. This establishes a baseline performance for the classification of release kinetics from nanomedicines.

An ML tool such as the one presented here could be applied for digital screening of experimental conditions and/or formulation properties to reduce material expenditure. With additional data and a larger dataset size, it is anticipated that the classifier accuracy would increase. Experimentalists could expand the dataset and this also highlights the benefit of adopting FAIR principles of scientific data management for published articles to the community.[44] Using existing, accessible data, this work maps the workflow for handling IVR and formulation data and provides an approach to de-risk experiments in often time critical pharmaceutical development activities. Overall, this research takes a step towards the accelerated development of non-oral dosage form IVR tests. This workflow could be adapted to other drug delivery modalities and demonstrates the potential benefit of adopting ML in pharmaceutical development to reduce experimental burden and streamline the transition of nanomedicine products into clinic.

Author contributions

Conceptualisation: DY, JT, MZ; data curation: DY; formal analysis and investigation: DY, JT, with contributions from all authors; methodology: DY, JT, MZ, MR, JM, HM, MP; software: DY; visualisation: DY, JT, MZ; writing – original draft: DY, MZ; writing – review & editing: all authors; supervision: VP, MR, JM, HM, MP, CA, JT, MZ; validation: DY, JT; project administration: JM, HM, MZ; funding acquisition: JM, MP, CA, JT, MZ.

Conflicts of interest

M. R., V. P., J. M., H. M. are employees of AstraZeneca and have stock ownership and/or stock options or interests in the company.

Data availability

The processed datasets and code are accessible in the GitHub (**https://github.com/danielyanes22/accelerated_IVR.git**) and Nottingham Research Data Management Repository (**https://doi.org/10.17639/nott.7522**).

Supplementary information containing methodologies for data curation, database construction and model evaluation as well as additional results on model outputs and performance is available. See DOI: **https://doi.org/10.1039/d5dd00112a**.

Acknowledgements

This project was supported by the Engineering and Physical Sciences Research Council EPSRC *via* the CDT in Transformative Pharmaceutical Technologies [EP/S023054/1]. Additionally, Daniel Vaughan (University of Nottingham, School of Pharmacy & Mathematical Sciences) is thanked for valuable discussions and advice. BioRender was used to create the graphical abstract (Created in BioRender. Yanes, D. (2025) **https://BioRender.com/h20f160**) and Fig. 1 (Created in BioRender. Yanes, D. (2025) **https://BioRender.com/p10b514**).

References

1 T. M. Allen and P. R. Cullis, *Adv. Drug Deliv. Rev.*, 2013, **65**, 36–48.
2 J. Shen and D. J. Burgess, *Drug Deliv. Transl. Res.*, 2013, **3**, 409–415.
3 L. Sercombe, T. Veerati, F. Moheimani, S. Y. Wu, A. K. Sood and S. Hua, *Front. Pharmacol.*, 2015, **6**, 286.
4 S. Pande, *Artif. Cells Nanomed. Biotechnol.*, 2024, **52**, 334–344.
5 Y. Svirkin, J. Lee, R. Marx, S. Yoon, N. Landrau, M. A. Kaisar, B. Qin, J. H. Park, K. Alam, D. Kozak, Y. Wang, X. Xu, J. Zheng and B. Rivnay, *Asian J. Pharm. Sci.*, 2022, **17**, 544–556.
6 H. Mead, V. Paraskevopoulou, N. Smith, R. Gibson, M. Amerio-Cox, G. Taylor-Vine, T. Armstrong, K. Harris, S. Wren and J. Mann, *Int. J. Pharm.*, 2023, 123317.
7 D. Solomon, N. Gupta, N. S. Mulla, S. Shukla, Y. A. Guerrero and V. Gupta, *AAPS J.*, 2017, **19**, 1669–1681.
8 L. Gómez-Lázaro, C. Martín-Sabroso, J. Aparicio-Blanco and A. I. Torres-Suárez, *Pharmaceutics*, 2024, **16**, 103.
9 M.-P. Mast, H. Modh, C. Champanhac, J.-W. Wang, G. Storm, J. Krämer, V. Mailänder, G. Pastorin and M. G. Wacker, *Adv. Drug Deliv. Rev.*, 2021, **179**, 113829.
10 L. P. Jahromi, M. Ghazali, H. Ashrafi and A. Azadi, *Heliyon*, 2020, **6**, e03451.
11 T. Lu and T. L. M. ten Hagen, *J. Controlled Release*, 2020, **324**, 669–678.
12 D. A. Rosser, B. R. Farris and K. C. Leonard, *Digit. Discov.*, 2024, **3**, 667–673.
13 S. Lu, B. Montz, T. Emrick and A. Jayaraman, *Digit. Discov.*, 2022, **1**, 816–833.
14 U. S. Vaitesswar, D. Bash, T. Huang, J. Recatala-Gomez, T. Deng, S.-W. Yang, X. Wang and K. Hippalgaonkar, *Digit. Discov.*, 2024, **3**, 210–220.
15 Z. Bao, J. Bufton, R. J. Hickman, A. Aspuru-Guzik, P. Bannigan and C. Allen, *Adv. Drug Deliv. Rev.*, 2023, **202**, 115108.
16 A. J. Gormley, *J. Controlled Release*, 2024, **373**, 23–30.
17 A. Ortiz-Perez, D. Van Tilborg, R. Van Der Meel, F. Grisoni and L. Albertazzi, *Digit. Discov.*, 2024, **3**, 1280–1291.

18 B. Muñiz Castro, M. Elbadawi, J. J. Ong, T. Pollard, Z. Song, S. Gaisford, G. Pérez, A. W. Basit, P. Cabalar and A. Goyanes, *J. Controlled Release*, 2021, **337**, 530–545.
19 P. Bannigan, Z. Bao, R. J. Hickman, M. Aldeghi, F. Häse, A. Aspuru-Guzik and C. Allen, *Nat. Commun.*, 2023, **14**, 35.
20 Y. Abdalla, L. E. McCoubrey, F. Ferraro, L. M. Sonnleitner, Y. Guinet, F. Siepmann, A. Hédoux, J. Siepmann, A. W. Basit, M. Orlu and D. Shorthouse, *J. Controlled Release*, 2024, **374**, 103–111.
21 B. Hoseini, M. R. Jaafari, A. Golabpour, A. A. Momtazi-Borojeni, M. Karimi and S. Eslami, *Sci. Rep.*, 2023, **13**, 18012.
22 R. Eugster, M. Orsi, G. Buttitta, N. Serafini, M. Tiboni, L. Casettari, J.-L. Reymond, S. Aleandri and P. Luciani, *J. Controlled Release*, 2024, **376**, 1025–1038.
23 Z. Bao, F. Yung, R. J. Hickman, A. Aspuru-Guzik, P. Bannigan and C. Allen, *Drug Deliv. Transl. Res.*, 2024, **14**, 1872–1887.
24 F. Marin, A. Rohatgi and S. Charlot, *arXiv*, 2017, preprint, arXiv:1708.02025, DOI: **10.48550/arXiv.1708.02025**.
25 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos and D. Cournapeau, *Journal of Machine Learning Research*, 2011, **12**, 2825–2830.
26 T. Chen and C. Guestrin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
27 S. M. Lundberg and S.-I. Lee, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2017, vol. 30.
28 R. Shwartz-Ziv and A. Armon, *Inf. Fusion*, 2022, **81**, 84–90.
29 G. Taylor-Vine, H. Mead, V. Paraskevopoulou and J. Mann, *Dissolution Technol.*, 2023, **30**, 126–132.
30 M. A. H. Capelle and T. Arvinte, *Drug Discov. Today Technol.*, 2008, **5**, e71–e79.
31 D. Reker, Y. Rybakova, A. R. Kirtane, R. Cao, J. W. Yang, N. Navamajiti, A. Gardner, R. M. Zhang, T. Esfandiary, J. L'Heureux, T. von Erlach, E. M. Smekalova, D. Leboeuf, K. Hess, A. Lopes, J. Rogner, J. Collins, S. M. Tamang, K. Ishida, P. Chamberlain, D. Yun, A. Lytoon-Jean, C. K. Soule, J. H. Cheah, A. M. Hayward, R. Langer and G. Traverso, *Nat. Nanotechnol.*, 2021, **16**, 725–733.
32 L. H. Lindner and M. Hossann, *Curr. Opin. Drug Discov. Devel.*, 2010, **13**, 111–123.
33 Y. Zhang, M. Huo, J. Zhou, A. Zou, W. Li, C. Yao and S. Xie, *AAPS J.*, 2010, **12**, 263–271.
34 F. Xie, S. Ji and Z. Cheng, *Eur. J. Pharm. Sci.*, 2015, **66**, 163–172.
35 S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal and S.-I. Lee, *Nat. Mach. Intell.*, 2020, **2**, 56–67.
36 J. Chen, D. Cheng, J. Li, Y. Wang, J. X. Guo, Z. P. Chen, B. C. Cai and T. Yang, *Drug. Dev. Ind. Pharm.*, 2013, **39**(2), 197–204.
37 M. Belkin, D. J. Hsu and P. Mitra, in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2018, vol. 31.
38 S. Georganos, T. Grippa, S. Vanhuysse, M. Lennert, M. Shimoni, S. Kalogirou and E. Wolff, *GIScience Remote Sens.*, 2018, **55**, 221–242.
39 P. M. Granitto, C. Furlanello, F. Biasioli and F. Gasperi, *Chemom. Intell. Lab. Syst.*, 2006, **83**, 83–90.
40 A. Mosavi, F. S. Hosseini, B. Choubin, M. Goodarzi and A. A. Dineva, *IEEE Access*, 2020, **8**, 145564–145576.
41 J. M. Johnson and T. M. Khoshgoftaar, *Journal of Big Data*, 2019, **6**, 27.
42 M. Ojala and G. C. Garriga, *Journal of Machine Learning Research*, 2010, **11**, 1833–1863.
43 P. Joyce, C. J. Allen, M. J. Alonso, M. Ashford, M. S. Bradbury, M. Germain, M. Kavallaris, R. Langer, T. Lammers, M. T. Peracchia, A. Popat, C. A. Prestidge, C. J. F. Rijcken, B. Sarmento, R. B. Schmid, A. Schroeder, S. Subramaniam, C. R. Thorn, K. A. Whitehead, C.-X. Zhao and H. A. Santos, *Nat. Nanotechnol.*, 2024, **19**, 1597–1611.
44 M. D. Wilkinson, M. Dumontier, Ij. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C. T. Evelo, R. Finkers, A. Gonzalez-Beltran, A. J. G. Gray, P. Groth, C. Goble, J. S. Grethe, J. Heringa, P. A. C. 't Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S. J. Lusher, M. E. Martone, A. Mons, A. L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M. A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao and B. Mons, *Sci. Data*, 2016, **3**, 160018.