Optimized Drug/Polymer Combination Prediction Using Machine Learning

Problem description

The problem addressed by this project is predicting and optimizing *in vitro* drug release behavior from PLGA (poly(lactide-co-glycolide)) nanoparticles. Although PLGA is one of the most widely used biodegradable polymers, formulation development remains largely empirical. Each new drug-polymer combination typically requires extensive experimentation to achieve a desirable sustained-release profile. With experimentally reported release data from 321 studies compiled in a PLGA microparticle dataset[1], it is possible to apply data-driven methods to identify relationships between formulation parameters, molecular descriptors, and concentration-time (drug release) outcomes.

The main question addressed here is whether machine learning and data analysis methods can accurately predict the relationships between formulation descriptors (such as particle size and drug loading efficiency) and the observed in vitro drug release kinetics. The hypothesis is that while linear correlations between these features and release outcomes are weak, multidimensional models may be able to capture the underlying interactions that govern release behavior. The goal is to predict how formulation design choices influence the duration and rate of drug release.

This problem is practically significant and benefits from the application of machine learning to a complex, multivariate system. It addresses a major bottleneck in drug formulation research: the inefficiency of experimental trial and error methods. If accurate clustering or predictive models can be developed, they can guide researchers toward favorable release kinetics based on known formulation features, reducing experimental cost and time. This approach also allows for a more systematic quantitative assessment of PLGA as a release-mediating polymer, accelerating the design of future sustained release systems.

Estimates for the cost associated with bringing a new drug product to market range from $172.5 million[1] to over $1.3 billion[2], when accounting for the costs associated with failed trials. The pharmaceutical industry is increasingly invested in computational drug discovery and drug product development approaches to minimize these failed trials (which account for over 90% of clinical efforts[3]). Accelerating the development pipeline in this way has positive financial implications for the pharmaceutical industry, and furthermore confers reduced cost and improved therapeutics to the end user[4].

Such computational approaches have been established for drug discovery of both small molecule[5] and ligand-based[4] drugs. Recent interest has developed applying such models to drug-excipient interactions[6,7]. However, to date, no publications exist directly modeling drug-eluting PLGA drug products, largely because of a lack of suitable datasets on which to train a machine learning model[1].

Data description

The dataset used in this project is an open-access dataset on formulation parameters and characteristics of drug-loaded poly(lactide-co-glycolide) (PLGA) microparticles, published by Bao *et al.* in March 2025[1]. It was compiled from 321 literatures, including 89 different drugs on their in vitro drug release studies. The data was first pooled from 1231 articles, and then reviewed by two researchers and verified on the collected data. Three more drug descriptors were added to the dataset using ‘RDKit’ library based on drugs’ SMILES string: molecular weight (kDa), topological polar surface area (TPSA), and LogP (a measure of drug’s lipophilicity). Apart from those three descriptors, polymer molecular weight (kDa), LA/GA ratio (ratio of lactic acid to glycolic acid, which correlates negatively with degradation rate)[9], initial drug-to-polymer ratio, particle size (µm), drug loading capacity (%), drug encapsulation efficiency (%), solubility enhancer concentration (%), time (days), and release (fractional drug release from PLGA MPs over time) were also reported in the dataset for potential analysis using machine learning applications.

The distribution of the drug release time in days and its release weight ratio is illustrated below in violin diagrams in **Fig.1**.

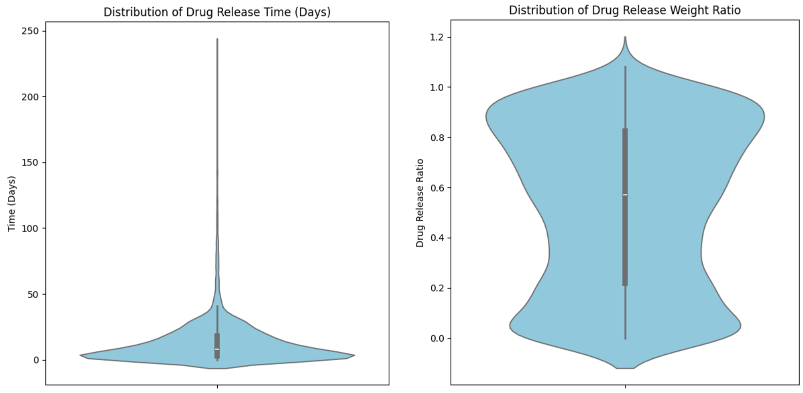

Fig.1: Violin plot illustration on the distribution of drug release time in days and drug release weight ratio

The drug release time has a median of 8.04 days, 1st quartile (Q1) 2.69 days, and 3rd quartile 18.08 days, with interquartile range (IQR) 15.39 days. This distribution indicates that most PLGA microparticle (MP) drugs are released within 20 days, with a small fraction exhibiting much longer release durations. The drug release weight ratio has a median of 0.57, Q1=0.22, Q3=0.82, IQR=0.61, which demonstrates a moderately symmetric distribution suggesting consistent release efficiency across different PLGA MP drug samples.

In short, while the distribution indicates that most drugs are released within 20 days, some exhibit prolonged release times, reflecting variability in release kinetics that is possibly influenced by polymer composition or formulation parameters.

Methods

The full microparticle release dataset was loaded from the Mendeley repository, containing 4913 rows and 13 columns. Each formulation was a block of rows, where each block represented a full release profile over time. The first step was to determine a shared release time that could be used as the regression target for a proof-of-concept model. This method would see if we can predict the fraction of drug released at a single, common time point purely from formulation descriptors. This keeps the problem constrained to a regression task, allowing us to test whether formulation variables actually contain a predictive signal for release behavior.

Initially, we attempted to identify a natural shared timepoint by rounding reported release times to the nearest tenth. However, rounding showed that no single time existed for every formulation, which was expected since the dataset was literature-mined from different studies that likely used different experimental parameters. To address this, we used cubic spline interpolation to interpolate the release curve for each formulation and evaluate the fraction released at a chosen target time, specifically for the formulations whose measured time range covered that point. This avoids artifacts from extrapolation and ensures that each interpolated value accurately reflects real experimental data. We initially selected t = 6 days because it was early enough to fall within the range of most profiles while still being physically meaningful. For every formulation whose time window included t = 6, we computed the interpolated release percentage value and constructed a new dataset containing one row per formulation. A small number of unrealistic values (four in total) were removed because they fell outside the valid fractional release bounds of 0 (no release) and 1 (complete release). After this cleaning, we were left with 301 usable formulations at the 6-day target time.

The SMILES strings provided with the Bao dataset were tokenized using a regex pattern popularized by DeepChem. This tokenizer split the strings into individual atoms, preserving bonds and placement relative to other atoms. These tokenizers were then passed to two featurizers: a fingerprint featurizer, and a physiochemical featurizer. A circular fingerprint of a molecule, also known as a Morgan fingerprint, places each atom at a specific locus within the number of bins provided to the featurizer (typically between 512 and 2048). This results in a vector of hundreds to thousands of features for each drug, spatially defining the placement of each atom relative to the others. It is known to be a fast and surprisingly robust featurizer, but poorly interpretable. The physiochemical featurizer, meanwhile, pulls 217 features of each molecule from DeepChem's database, with parameters related to charge, polarity, size, and shape, among others. This creates a 217-feature-long vector for each drug. These features tend to result in slower machine learning performance but more interpretable results. We reserved these SMILES-based features to compare against the base features provided in the dataset, after tuning each of our models to their best possible performance as described below.

We then removed any columns that were not true formulation descriptors, including the time column, the interpolated release column (our target), and the formulation index identifier. This resulted in 10 usable formulation features, all of which were numeric and contained no missing values. Before supervised learning, we applied a Standard Scaler to normalize the data. This was necessary since the features varied significantly in magnitude and physical units. Without scaling, features with inherently larger numeric ranges would dominate the model's optimization steps. We then performed unsupervised analysis with Principal Component Analysis (PCA) to understand the structure and variance in the formulation feature space. The explained-variance plot showed that we would need 8 principal components to retain 95% of the variance and 5 principal components to retain 75%. Since training time was not a limiting factor, we decided to keep all features for modeling rather than reduce dimensionality. A PCA loadings heatmap confirmed the relative contributions of each formulation descriptor to the principal components.

For supervised learning, we began with a linear regression model with an L2 (Ridge) penalty. Since our goal was to predict continuous numerical release values, the problem naturally required a regression model rather than a classification model. The linear model was used to illustrate how poorly it handles drug-release kinetics. PLGA release involves nonlinear diffusion, polymer erosion, and various physical interactions that cannot be captured by a simple weighted sum of features.

We then used a Support Vector Regression (SVR) to model drug release behavior. SVR, a regression extension of Support Vector Machines (SVM), uses kernel functions to capture nonlinear relationships by projecting input variables into a higher-dimensional feature space. In this work, a radial basis function (RBF) kernel was used, allowing the model to learn smooth, nonlinear mappings between formulation descriptors and release percentages.

Next, we applied a Random Forest regressor. Random Forest was chosen because it is a nonlinear ensemble model composed of many decision trees, each of which performs repeated feature splits and yields piecewise constant regions. The model has the ability to approximate nonlinear relationships and interactions in the PLGA dataset. We trained it using 5-fold cross-validation to assess its generalization. After establishing a baseline, we performed hyperparameter optimization using GridSearchCV over a range of depths, estimators, and split strategies. This let us evaluate model performance and identify an optimally tuned configuration. Finally, we repeated the entire interpolation process across multiple target times (6, 8, 10 days, etc.) to see if the choice of shared timepoint affected predictive power. This was motivated by the fact that very early release readings in longer release profile samples can be noisy due to measurement sensitivity and instrument limitations, while slightly later timepoints may reflect more stable kinetics.

To extend the framework beyond single-timepoint prediction toward full kinetic modeling, we explored fitting an empirical release model to each formulation and then training machine learning models to predict the fitted parameters directly. This approach enables reconstruction of an entire release curve from formulation descriptors alone, rather than predicting release at a single time. The empirical model selected was the Weibull release model, which is commonly used to describe drug release kinetics from polymeric systems. The Weibull model depends on time as the independent variable and includes three fitted parameters: an amplitude term (A), which represents the asymptotic maximum fraction released, a scale parameter (tau), which sets the characteristic timescale of release, and a shape parameter (Beta), which controls the curvature of the release profile.

For each formulation, the Weibull model was fit to the experimentally reported release profile, yielding a unique set of parameters describing that formulation's release kinetics. While the Weibull model provides an excellent descriptive fit to individual release curves, it is not predictive on its own, as the parameters must be obtained from experimental release data. To introduce predictive capability, machine learning models were trained to map formulation descriptors to the fitted Weibull parameters. In this framework, a new formulation could be evaluated by first predicting its Weibull parameters from formulation features and then reconstructing the full release profile using the Weibull equation.

To test this approach, three supervised regression models were evaluated: a ridge regression model as a linear baseline, a random forest regressor to capture nonlinear feature interactions, and a multilayer neural network with seven hidden layers to assess whether deeper architectures could learn more complex mappings. The dataset consisted of the complete 321 formulations, each represented by formulation descriptors as inputs and the corresponding Weibull parameters as targets. The data were split into isolated training and testing sets to evaluate generalization performance and avoid information leakage.

Results and Conclusion

The baseline linear model with the L2 penalty performed as expected for a system governed by complex kinetics. We used R^2 as the primary evaluation metric because it measures how much of the variance in release behavior is explained by the model, indicating how well our model fits the data. The Ridge Regression model achieved an R^2 value of 0.2820 on an isolated test set, confirming that linear relationships are insufficient for capturing the complexity of PLGA release behavior (**Fig. 2**). This justified exploring nonlinear approaches.

SVR performed poorly, yielding an R^2 value of 0.244. This reflects its limited ability to interpret the drug release curves at a shared timepoint of $t = 6$ days. The Random Forest regressor performed substantially better. With 5-fold cross-validation, it achieved an R^2 of 0.4577 and a Mean Average Error (MAE) of 0.1369 (**Fig. 2**). This confirmed that the Random Forest (RF) model successfully captured more meaningful patterns in formulation-release relationships compared to the linear model and support vector regressor. Feature importance analysis showed that drug encapsulation efficiency and polymer molecular weight were the top contributors. This aligned with our expectations, as the polymer molecular weight controls the degradation rate, and encapsulation efficiency reflects the drug density in the nanoparticle, which in turn affects particle size and the diffusion path length.

Hyperparameter tuning with GridSearchCV evaluated 108 model configurations (540 fits total after 5-fold cross-validation). The best model achieved an R^2 of 0.4241 and an MAE of 0.1404 when applied to the test set. Notably, hyperparameter tuning did not outperform the baseline RF model, suggesting that the initial parameterization was already near a reasonable optimum given the structure of the dataset.

Because t = 6 days might have been too early to draw conclusions from, and early-stage releases can suffer from experimental noise, we evaluated multiple candidate target times by re-interpolating the dataset at each time and running 5-fold Cross-Validation. The best model performance occurred at t = 10 days, where the model achieved an R^2 value of 0.5220 and an MAE of 0.158 (**Fig. 2**). The R^2 value was noticeably higher than the t = 6 results, reinforcing the idea that t = 6 days may not have been the optimal choice for predicting formulation characteristics.

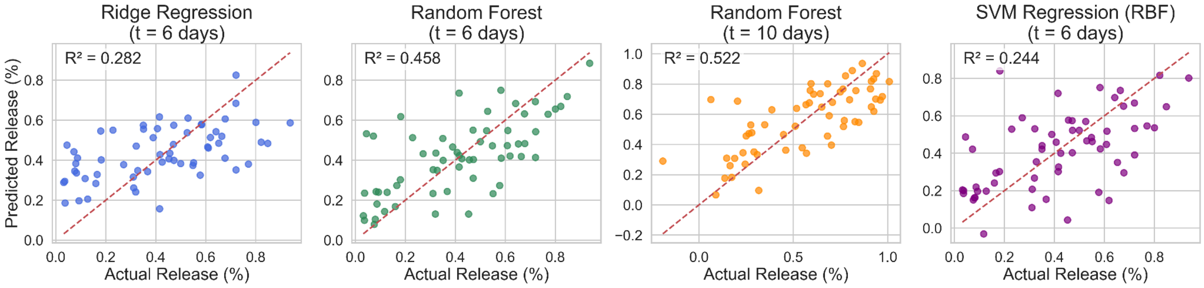

Fig. 2: Predicted vs. actual drug release percentages for three regression models evaluated on an isolated testing dataset.

The next step was applying a new methodology to model the full release profiles over time. The Weibull model was successfully fit to all 321 formulation release profiles and showed excellent descriptive performance. The average R^2 across all fits was 0.9810, with values ranging from 0.7153 to 1.000, confirming that the three-parameter Weibull model accurately captures the shape and kinetics of individual release curves when fit directly to experimental data. Despite this strong descriptive performance, predicting the fitted Weibull parameters from formulation descriptors proved difficult. Ridge regression performed poorly, with a maximum R^2 of 0.017 for all three parameters, indicating negligible linear relationships. Random forest regression provided limited improvement, achieving a maximum R^2 of approximately 0.197 for a single parameter, while the seven-layer neural network failed to generalize, yielding a best R^2 of -0.167. These results indicate that although the Weibull model fits experimental release profiles extremely well, its fitted parameters are not reliably predictable from the 10 formulation descriptors alone across the full dataset (**Fig. 3**). This suggests that release kinetics are influenced by factors not fully captured by the 10 specific formulation parameters.

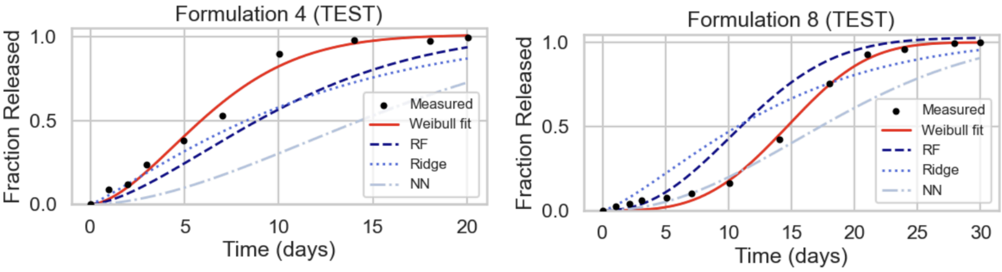

Fig. 3: Experimental release profiles (black points) compared with direct Weibull fits (red) and release curves reconstructed from Weibull parameters predicted by random forest, ridge regression, and neural network models for 2 test-set formulations.

Testing the optimized random forest model using the fingerprint and physiochemical features acquired from the SMILES strings reveals surprisingly little improvement in overall performance. Figure 3 below shows the number of (overall and most important) features in each dataset, and the R^2 values achieved by the models at t=6 days and t=10 days.

Dataset	Total Features	Features Required for 95% Variance Explanation
Base (no SMILES features)	10	8
Base + Fingerprint	227	34
Base + Physiochemical	1034	66

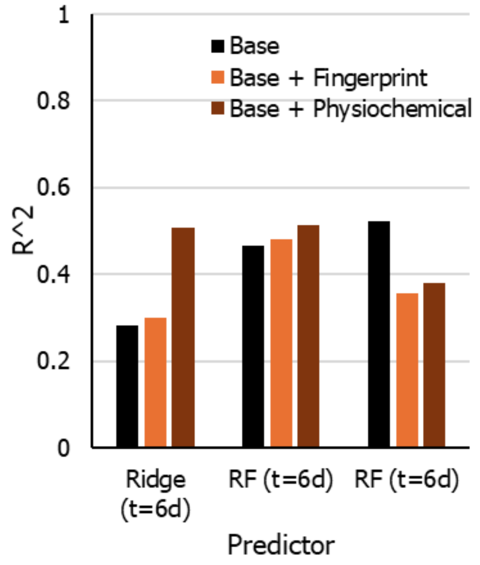

Fig 4: (left) Incorporating features based on the SMILES strings of each drug increases the number of features required to explain the variance in the data. (right) Models based on SMILES features outperform those based on only the Bao features at short timepoints, but at longer timescales the core features prevail in importance.

As noted earlier under "Methods," circular/Morgan fingerprinting sacrifices interpretability for rapid model performance. While loci 169, 887, and 944 (of the 1024 we offered to the featurizer) were identified as contributing most strongly to the model's predictions, there is no simple way to extract which atoms lay at these loci, nor how they relate spatially to the rest of the molecule. On the other hand, several more interpretable physiochemical features were identified: the molecular flexibility index ϕ, the Balaban J index for molecular connectivity, hydrophobicity, van der Waals surface area, electron-richness of certain atoms, and molecular polarizability. When viewing these features in the context of drug-polymer interactions, it is easy to imagine how these picomechanical and electronic properties might play a role in determining a drug's ability to interact with the PLGA matrix and dissolve in an aqueous environment.

It is worth noting here, however, that the improvements in the RF model at predicting release behavior at a single timepoint are, at best, marginal; overall model performance is still poor, with the maximum R^2 values achieved hovering around 0.5.

Discussion and Future Work

Overall, these results show that formulation descriptors may be meaningful predictors of release behavior, but only when nonlinear models are applied. Linear regression and Support Vector Regression were unable to capture these relationships, while Random Forests handled the complexity more appropriately. The single-timepoint system was satisfactory in the sense that the model consistently identified physically meaningful features as dominant predictors, reinforcing that an underlying learning signal may exist. However, Random Forests still underperformed relative to expectations, and the same limitation was observed for the kinetic modeling approach, where machine learning models were unable to reliably predict Weibull parameters from formulation descriptors alone, despite the empirical model providing excellent descriptive fits to the release profiles.

It is important to note that this dataset was compiled from hundreds of studies, each with potentially differing laboratory conditions, measurement protocols, release media, and instrumental sensitivities. The variations between laboratories introduce unavoidable noise in the profiles, which could have affected the cubic spline interpolation approximations of shared timepoints.

The largest area for improvement is the incorporation of additional contextual information, such as polymer chemistry details, or processing conditions, to reduce noise and constrain parameter learning. Future work should focus on integrating richer formulation and processing metadata, exploring hybrid physics-informed machine learning approaches, and evaluating alternative representations of kinetics that may be more directly learnable.

References

(1) Bao, Z.; Kim, J.; Kwok, C.; Le Devedec, F.; Allen, C. A Dataset on Formulation Parameters and Characteristics of Drug-Loaded PLGA Microparticles. *Sci. Data* **2025**, *12* (1), 364. https://doi.org/10.1038/s41597-025-04621-9.

(2) Sertkaya, A.; Beleche, T.; Jessup, A.; Sommers, B. D. Costs of Drug Development and Research and Development Intensity in the US, 2000-2018. *JAMA Netw. Open* **2024**, *7* (6), e2415445. https://doi.org/10.1001/jamanetworkopen.2024.15445.

(3) Wouters, O. J.; McKee, M.; Luyten, J. Estimated Research and Development Investment Needed to Bring a New Medicine to Market, 2009-2018. *JAMA* **2020**, *323* (9), 844–853. https://doi.org/10.1001/jama.2020.1166.

(4) Sun, D.; Gao, W.; Hu, H.; Zhou, S. Why 90% of Clinical Drug Development Fails and How to Improve It? *Acta Pharm. Sin. B* **2022**, *12* (7), 3049–3062. https://doi.org/10.1016/j.apsb.2022.02.002.

(5) Sadybekov, A. V.; Katritch, V. Computational Approaches Streamlining Drug Discovery. *Nature* **2023**, *616* (7958), 673–685. https://doi.org/10.1038/s41586-023-05905-z.

(6) Sabe, V. T.; Ntombela, T.; Jhamba, L. A.; Maguire, G. E. M.; Govender, T.; Naicker, T.; Kruger, H. G. Current Trends in Computer Aided Drug Design and a Highlight of Drugs Discovered via Computational Techniques: A Review. *Eur. J. Med. Chem.* **2021**, *224*, 113705. https://doi.org/10.1016/j.ejmech.2021.113705.

(7) Antipas, G. S. E.; Reul, R.; Voges, K.; Kyeremateng, S. O.; Ntallis, N. A.; Karalis, K. T.; Miroslaw, L. System-Agnostic Prediction of Pharmaceutical Excipient Miscibility via Computing-as-a-Service and Experimental Validation. *Sci. Rep.* **2024**, *14* (1), 15106. https://doi.org/10.1038/s41598-024-65978-2.

(8) Dignon, G. L.; Dill, K. A. Computational Procedure for Predicting Excipient Effects on Protein–Protein Affinities. *J. Chem. Theory Comput.* **2024**, *20* (3), 1479–1488. https://doi.org/10.1021/acs.jctc.3c01197.

(9) Panigrahi, D., Sahu, P.K., Swain, S. et al. Quality by design prospects of pharmaceuticals application of double emulsion method for PLGA loaded nanoparticles. SN Appl. Sci. 3, 638 (2021). https://doi.org/10.1007/s42452-021-04609-1

(10) Yamada, H.; Liu, C.; Wu, S.; Koyama, Y.; Ju, S.; Shiomi, J.; Morikawa, J.; Yoshida, R. Predicting Materials Properties with Little Data Using Shotgun Transfer Learning. *ACS Cent. Sci.* **2019**, *5* (10), 1717–1730. https://doi.org/10.1021/acscentsci.9b00804.