ACS OMEGA

http://pubs.acs.org/journal/acsodf

Article

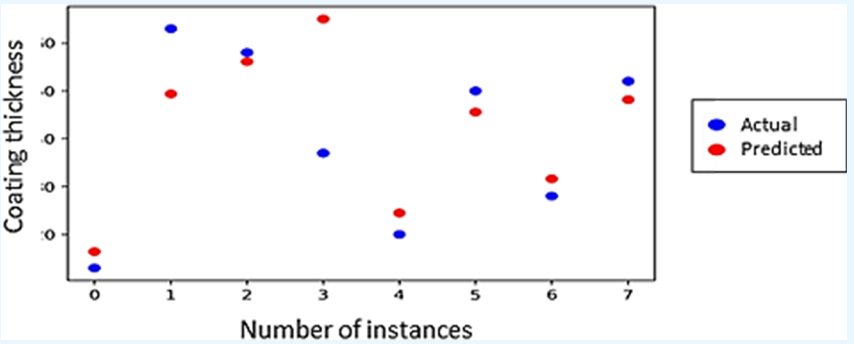
The Effect of Machine Learning Algorithms on the Prediction of Layer-by-Layer Coating Properties

Tijana Šušteršič, Varvara Gribova, Milica Nikolic, Philippe Lavalle, Nenad Filipovic, and Nihal Engin Vrana*

Cite This: *ACS Omega* 2023, 8, 4677−4686

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information



ABSTRACT: Layer-by-layer film (LbL) coatings made of polyelectrolytes are a powerful tool for surface modification, including the applications in the biomedical field, for food packaging, and in many electrochemical systems. However, despite the number of publications related to LbL assembly, predicting LbL coating properties represents quite a challenge, can take a long time, and be very costly. Machine learning (ML) methodologies that are now emerging can accelerate and improve new coating development and potentially revolutionize the field. Recently, we have demonstrated a preliminary ML-based model for coating thickness prediction. In this paper, we compared several ML algorithms for optimizing a methodology for coating thickness prediction, namely, linear regression, Support Vector Regressor, Random Forest Regressor, and Extra Tree Regressor. The current research has shown that learning algorithms are effective in predicting the coating output value, with the Extra Tree Regressor algorithm demonstrating superior predictive performance, when used in combination with optimized hyperparameters and with missing data imputation. The best predictors of the coating thickness were determined, and they can be later used to accurately predict coating thickness, avoiding measurement of multiple parameters. The development of optimized methodologies will ensure different reliable predictive models for coating property/function relations. As a continuation, the methodology can be adapted and used for predicting the outputs connected to antimicrobial, anti-inflammatory, and antiviral properties in order to be able to respond to actual biomedical problems such as antibiotic resistance, implant rejection, or COVID-19 outbreak.

1. INTRODUCTION

Implants and prostheses have become common routes to treat various medical pathologies. However, their use is often linked to complications such as an averse immune response and bacterial and fungal infections.[1,2] In this context, new approaches are urgently needed. However, development of new biomaterial systems empirically entails significant costs and time consumption.[3,4]

One of the approaches to modify implant surfaces consists of layer-by-layer (LbL) film deposition, which is very versatile and can be used for multiple biomedical applications[5,6] via antibacterial and anti-inflammatory films,[7−9] osteogenic films,[10,11] drug delivery microcapsule fabrication,[12] etc. Other applications include food packaging,[13] optical fiber sensing,[14] and many electrochemical systems.[15]

However, despite the number of publications in the field,[16] predicting film properties represents quite a challenge, and variations from bulk behaviors are common. Film composition, its component properties, ratios, pre-processing, environmental factors, and a variety of other material-based parameters have also been found to have a significant effect on film properties.[17,18] Polymer manufacturing is difficult to forecast and model in general owing to the interdependence of processing environments, polymer structures, and geometries.

Received: October 7, 2022
Accepted: December 30, 2022
Published: January 30, 2023

Table 1. Dataset Inputs for Predicting Film Thickness (nm)[a]

input name	description (continuous/categorical/yes−no) variable				
method	OWLS	AFM−CLSM	AFM	QCM-D	profilometry
	OFAR	ellipsometry	SEM	NR	
polycation	PAH	PEI	CHI	PLL	PDADMA
	PDADMAC	glyc-CHI	COL	PTEMC	PAni
	iMAPA	PAR			
polyanion	PGA	HA	PAA	PSS	ALG
	CSA	HEP	PCBS	PSSMA	FUC
	CARiota	Pectin	DEX	CARkappa	CARlambda
polycation unit MW, kDa	continuous variable				
polyanion unit MW, kDa	continuous variable				
polycation MW, kDa	continuous variable				
polyanion MW, kDa	continuous variable				
number of bilayers	continuous variable				
ending polymer	none	PGA	HA	PAA	PSS
	PDADMA	CSA	glyc-CHI	PCBS	HEP
	PAH	CARiota	pectin	COL	CHI
	PLL	PEI	FUC	CARkappa	CARlambda
concentration of polyanion, mg/mL	continuous variable				
concentration of polycation, mg/mL	continuous variable				
cross-linking	no	EDC/S-NHS		EDC/NHS	
	GLUT	genipin		glutaraldehyde	
duration of each layer deposition, min	continuous variable				
presence of negatively charged groups: COO^-	yes/no				
charge density (charges per unit), polycation	continuous variable				
charge density (charges per unit), polyanion	continuous variable				
presence of negatively charged groups sulfonates: $-SO_3^-$	yes/no				
presence of negatively charged groups sulfates: $-O-SO_3^-$	yes/no				
pH polycation	continuous variable				
pH polyanion	continuous variable				
buffer	no value	HEPES	MES + Tris	Tris	NaCl
	KCl	$MgCl_2$	acetate	$NaNO_3$	
buffer concentration, mM	continuous variable				
NaCl, M	continuous variable				

[a]AFM: atomic force microscopy; ALG: alginate; CARiota: ι-carrageenan; CARkappa: κ-carrageenan; CARlambda: λ-carrageenan; CHI: chitosan; CLSM: confocal laser scanning microscopy; COL: collagen; CSA: chondroitin sulfate; DEX: dextran; EDC: 1-ethyl-3-(3-dimethylaminopropyl)-carbodiimide; FUC: fucoidan; GLUT: glutaraldehyde; glyc-CHI: glycol-chitosan; HA: hyaluronic acid; HEP: heparin; iMAPA: insoluble multi-L-arginyl-poly-L-aspartate; NR: neutron reflectometry; OFAR: optical fixed angle reflectometry; OWLS: optical waveguide light-mode spectroscopy; PAA: poly(acrylic acid); PAH: poly(allylamine hydrochloride); PAR: poly(L-arginine); PAni: polyaniline; PCBS: poly[1-[4-(3-carboxy-4-hydroxyphenylazo)benzenesulfonamido]-1,2-ethanediyl, sodium salt; PDADMA: poly(diallyldimethylammonium chloride); PDADMAC: poly(diallyldimethylammonium chloride); PEI: polyethylenimine; PGA: poly(L-glutamic acid); PLL: poly(L-lysine); PSS: poly (styrene sulfonate); PSSMA: poly(4-styrenesulfonic acid-co-maleic acid); PTEMC: poly(trimethylammonium ethyl methacrylate chloride); QCM-D: quartz crystal microbalance with dissipation monitoring; SEM: scanning electron microscopy; S-NHS: N-hydroxysulfosuccinimide.

Process modeling usually entails large amounts of testing as well as computational modeling and/or numerical simulation, and such techniques are often cyclical, with many trial-and-error runs, expensive and time-consuming experiments, and material losses.[19,20]

Machine learning (ML) algorithms have been advocated to solve complex modeling and optimization problems in numerous engineering fields as numerical computational power has increased.[21] It is a subfield of artificial intelligence (AI) concerned with the creation of models (knowledge) that can efficiently learn from real data.[22−24] Over the last few decades, ML has evolved into a wide field of study, resulting in a variety of different algorithms, hypotheses, methods, implementation areas, etc.[25] However, learning/algorithms have been broadly classified into three categories:

- supervised learning, in which learning is dependent on comparing computed output to desired output; the algorithm creates a model that maps inputs to desired outputs;
- unsupervised learning, in which learning is based solely on the pattern of input; the algorithm is programmed to derive structure from results;
- reinforcement learning, in which the algorithm learns policies/rules on how to behave in order to produce the best outcomes by trial and error.[25,26]

Today, the field of ML has proven useful in many industries and scientific fields. ML algorithms have shown great promise as effective methods for simulation and classification of dynamic manufacturing processes[27] and materials science problems.[28−30] When compared to traditional statistical modeling techniques like linear regression and response surface methodology, ML-based approaches have shown dominance as modeling techniques for data sets with nonlinear relationships.[31,32] These techniques have demonstrated surprising capability in recognizing patterns in complex systems

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

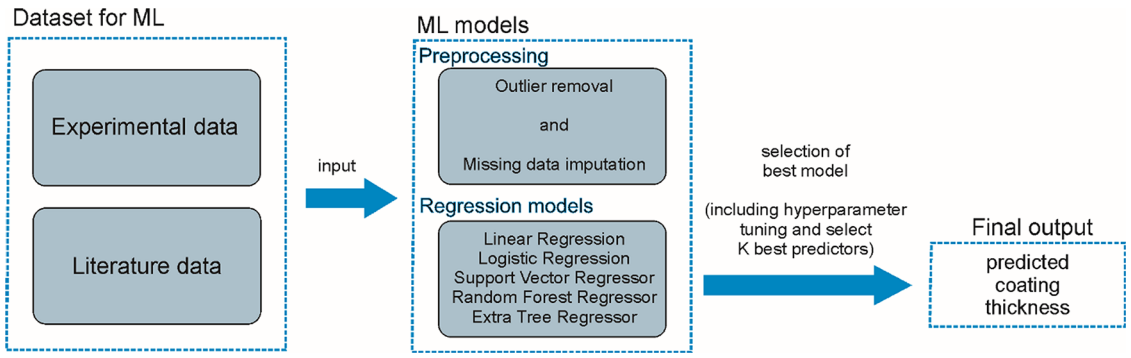


Figure 1. Workflow for coating thickness prediction using machine learning.

and capture interactions among input and output variables in a system. They have also shown enormous performance in quantitative structure−property relationship investigations.[28] In the field of LbL films, which often take a long time to fabricate, ML algorithms are promising to solve a problem of film property prediction, from their physicochemical to biological features. This would save time and allow accelerated and improved LbL film development for various applications as the production of well-controlled (nanoscale precision) LbL films of 1 μm and higher can take at least several hours for each pair of polyelectrolytes studied. However, up to now, the use of ML is limited in the biomaterials field due to the lack of normalized data, the multiparametric nature of the structure function relationships, and also the lack of industrial processes that are comparable in their nature.

As an iterative process, where parametrization is easily achievable, LbL coatings provide an opportunity to transfer ML techniques to functional biomaterial design. As such, in the recent paper by Gribova et al., the authors used literature data and in-house generated experimental results to analyze the relative impact of 23 different coating parameters on the coating thickness.[33] This is the first time the authors utilized ML in prediction of coating properties. However, in order to simplify the modeling process, Gribova et al. assumed that coating thickness has a linear relationship with the number of bilayers. As a result, they have adopted eight bilayers as a basis for calculation of coating thickness. On the contrary, in this paper, we do not use a constant number of bilayers as input, but rather a number of bilayers in the range of 1 to 125 is taken as one of the features in prediction. Additionally, in this paper, we develop a methodology for prediction of coating output based on several ML algorithms, namely, linear regression, Support Vector Regressor (SVR), Random Forest Regressor, and Extra Tree Regressor. These algorithms do not assume linearity and moreover include also nonlinear relationship with input attributes. In general, nonlinear algorithms have many advantages that make them ideal for forecasting regression problems in a multiparameter production process.[31] Following the demonstration of the feasibility of the coating property prediction, it is important to analyze the suitability of the available algorithm options to ensure models with the least overfitting and better predictive capacities.

The main contributions of the paper are reflected in several aspects:

- ML algorithm selection is employed to the problem of coating thickness prediction, utilizing advanced methodology for a problem that was not approached in such a way before;
- reduction of number of experiments in finding the optimal conditions (thickness of different coatings is hard to control as they are a factor of many attributes, thus leading to large material consumption and repetitive experiments);
- predictive nature of the model that allows for fast response in different applications (i.e., response to the COVID-19 pandemic).

2. MATERIALS AND METHODS

2.1. Data Collection. The methodology for data collection, both from the literature and experimental data, was previously described in the paper by Gribova et al.[33] Briefly, film thickness data from the literature were combined with the in-house data produced using a Quartz Crystal Microbalance with dissipation monitoring (QCM-D), and different features such as polymer concentration, molecular weight, charge density, etc., were included (Table S1).

2.2. Dataset. The whole dataset for prediction of coating thickness included the 22 input features presented in Table 1 (method as a feature was excluded from analysis, so the used attributes were polycation, polyanion, polycation unit MW, polyanion unit MW, polycation MW, polyanion MW, number of bilayers, ending polymer, concentration of polyanion, concentration of polycation, cross-linking, duration of each layer deposition, etc.). Targeted prediction output is film thickness (measured by QCM-D, CLSM etc.) in nm. The total number of instances was 98 from the literature and 33 from the in-house experiments, resulting in a total of 131 instances.

Description of the dataset in the form of mean, standard deviation, and minimal and maximal values is given:

- Mean of film thickness is 677 nm;
- Standard deviation of film thickness is 2051 nm;
- Minimum of film thickness is 3 nm;
- Maximum of film thickness is 15000 nm.

In the preprocessing stage, label encoding was used to convert each categorical value in a column to a number, based on a total number of unique values (i.e., polycation had 12 different values, meaning that these strings were coded as numbers 1−12).

2.3. Workflow. The estimation of coating thickness was based on 22 different input features, after which the machine learning methods were introduced to employ automatic methods for prediction of coating thickness and determine the most relevant parameters in output thickness prediction. The workflow for the proposed methodology is given in Figure 1. Combined experimental and literature data are forwarded as

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

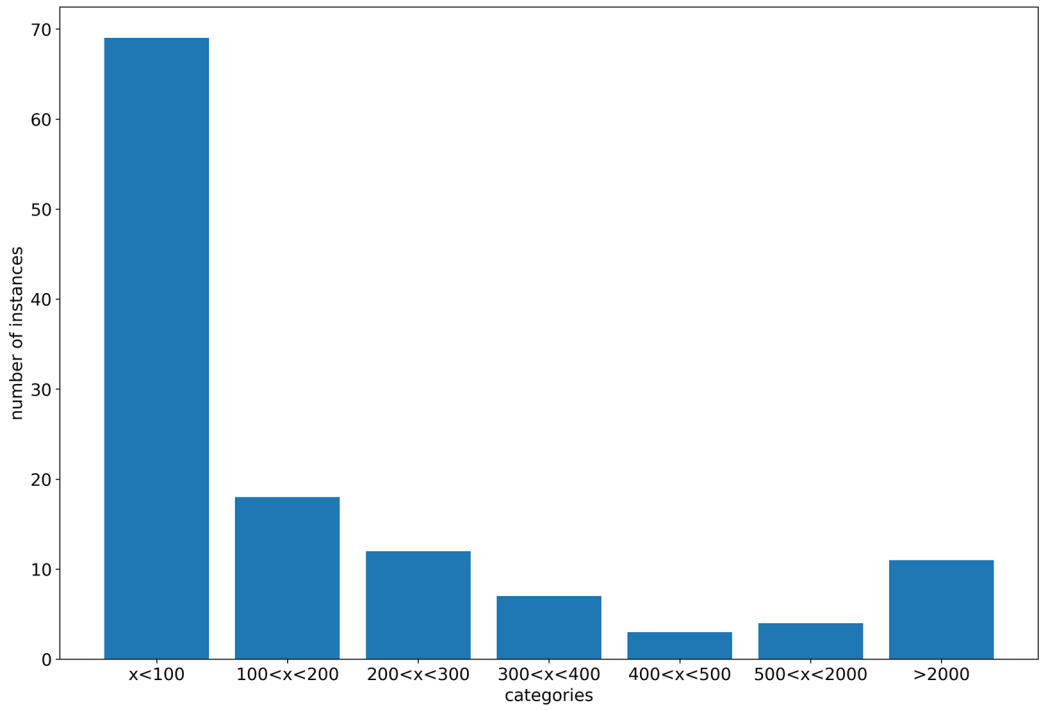


Figure 2. Distribution of film thickness (nm) categories.

inputs to the ML models. As preprocessing methods, outlier removal and missing data imputation are performed, after which different regression models are employed (including hyperparameter tuning and selection of K best predictors of the output) to get the final output − predicted coating thickness.

2.4. Machine Learning. Several linear and nonlinear regression models are implemented to forecast the coating thickness as a result of the 23 predictors. Standard division of the whole dataset into training, validation, and test subsets is performed. Randomized presentation of the instances in the batches is also used to avoid overfitting.

2.4.1. Regression Algorithms. Linear regression is probably one of the most widely used regression methods. One of its main advantages is the ease of interpreting the results. When implementing linear regression of a dependent variable y based on a set of independent variables $x = (x_1, ..., x_r)$, where r is the number of predictors, a linear relationship between y and x is assumed in the form

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \varepsilon \quad (1)$$

This equation contains $\beta_0, \beta_1, ..., \beta_r$, which are the regression coefficients, and ε is the random error. The method works by calculating the estimators of the regression coefficients. The estimated or predicted response, $y = f(x_i)$ for each observation $i = 1, ..., n$, should be as close as possible to the corresponding actual response y_i, in our case experimental or literature value for coating thickness. The differences $y_i - f(x_i)$ for all observations are called the residuals. The purpose of regression is to determine the best-predicted weights, meaning with the smallest residuals. This is achieved by minimizing the sum of squared residuals (SSR) for all observations:

$$SSR = \Sigma_i (y_i - f(x_i))^2 \quad (2)$$

This approach is called the method of ordinary least squares.[34]

Support Vector Regression (SVR) uses the same principle as the Support Vector Machine (SVM) but for regression problems. The goal is to find the hyperplane that represents the decision boundary. Further, the points that are within the decision boundary line are considered and we should determine the best fit line (hyperplane) that has a maximum number of points. In contrast to Ordinary Least Squares (OLS), the objective function of SVR is to minimize the coefficients—more specifically, the l2-norm of the coefficient vector—not the squared error. The error term is instead handled in the constraints, where we strive toward the absolute error less than or equal to a specified margin, called the maximum error, ϵ (epsilon). Epsilon is tuned to gain the desired accuracy of the model.[35]

For the case of predicting a numeric variable, where linear models produced not so great results, "model trees" are proposed, meaning decision tree regressors. Decision tree regression evaluates an object's attributes and trains a model with a tree structure to forecast data in the future to create meaningful continuous output.

A Random Forest (RF) is a meta estimator that fits a number of regression decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. Because there are several trees and each tree is trained on a portion of data, the Random Forest approach is less biased. Essentially, the Random Forest algorithm depends on the strength of "the crowd," which reduces the system's overall bias. Another advantage of the algorithm is that it is very stable, meaning that if a new data point is introduced in the dataset, the overall algorithm is not affected much since new data may impact one tree, but it is very hard for it to impact all the trees. The Random Forest algorithm has also proven well when the dataset contains both categorical and numerical features, which is the case in our dataset. The Random Forest algorithm also works well when

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

data has missing values or it has not been scaled well, proving it also suitable to implementation in our problem.[36]

Another advanced algorithm is the Extra Tree algorithm, which is an ensemble of decision trees and is related to other ensembles of decision trees algorithms such as bootstrap aggregation (bagging) and Random Forest. The Extra Tree algorithm works by creating a large number of unpruned decision trees from the training dataset. Predictions are made by averaging predictions from decision trees in the case of regression.[37]

While described model parameters are learned during training, i.e., slope and intercept in a linear regression, hyperparameters must be initialized before training. In the case of a Random Forest, hyperparameters include, i.e., number of decision trees in the forest and the number of features considered by each tree when splitting a node. Scikit-Learn in Python implements a set of sensible default hyperparameters for all models, but these are not guaranteed to be optimal for a problem. The best hyperparameters are usually impossible to determine ahead of time, and tuning a model is necessary. Among many methods for hyperparameter tuning, we adopted RandomizedSearchCV to create a parameter grid to sample from during fitting. In such a way, a grid of hyperparameter ranges is defined and randomly sampled from the grid, performing K-Fold CV with each combination of values.[38]

2.4.2. Outliers Removal and Selection of K Best Features. Besides implementing the specific models and their optimization, we investigated outlier removal, in order to improve the model accuracy. An outlier is a data point whose response y does not follow the general trend of the rest of the data. A data point has high leverage if it has ″extreme″ predictor x values. With a single predictor, an extreme x value is simply one that is particularly high or low. With multiple predictors, extreme x values may be particularly high or low for one or more predictors or may be ″unusual″ combinations of predictor values (e.g., with two predictors that are positively correlated, an unusual combination of predictor values might be a high value of one predictor paired with a low value of the other predictor). In statistics, Cook's distance is a common measurement of a data point's influence. It is a way to find influential outliers in a set of predictor variables when performing a least-squares regression analysis. Therefore, we have investigated the leverage vs studentized residuals plot to detect any outliers.

As part of the proposed methodology and investigation of the distribution of film thickness categories, it was seen that more than half of instances (69) had the coating thickness less than 100 nm (Figure 2). As a result, we will investigate if there are potentially outliers that can be removed from the dataset in order to improve the prediction.

Additionally, the Select K best method, which selects features according to the k highest scores, is used to determine the K best predictors of the output.[39]

2.4.3. Software and Libraries. The ML model for coating thickness prediction is implemented in Python 3.7, Spyder environment, using Scikit-Learn version 0.23.2 and Numpy 1.16.5.

3. RESULTS AND DISCUSSION

As described in the methodology section, several ML models have been tested to determine which regression model achieves the highest results. The results are given in the form of R^2, mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE). For regression models, R-squared-type goodness-of-fit summary statistics have been constructed for particular models using a variety of methods.[40] The results are reported based on 5-fold cross validation. The findings on the test dataset are given in Table 2. It should be emphasized that these results are based on data before any outlier removal investigation.

Table 2. Results from Regression Analysis Applied to Prediction of Coating Thickness on an Original Test Set

	Test subset			
Regression model	R^2 score	MSE [nm]	RMSE [nm]	MAE [nm]
Linear regression	0.77	645691.52	803.55	485.94
Support Vector Regressor	−0.17	3246351.92	1801.76	736.22
Random Forest Regressor	0.85	419388.84	647.60	318.13
Extra Tree Regressor	0.92	242290.20	492.22	237.5

Extra Tree Regressor proved to be the best during training and testing, achieving R^2 score on a test set of 0.92, MSE of 242290.20 [nm], RMSE of 492.22 [nm], and MAE of 237.5 [nm].

In order to remove the outliers, the leverage vs studentized residuals plot was investigated (Figure 3) and noticed that the instance number 3 could be declared as an outlier (small leverage and high residual).

After removal of instance 3 (Figure 4), the accuracy of the model improved as well as R^2, increasing from 0.92 to 0.98 on a test set.

After outlier removal, a standard estimation of missing values (data imputation) was applied in order to determine if data imputation will increase the model accuracy. A total of 602 values were missing (number of missing values per attribute were 0, 0, 3, 7, 9, 25, 0, 12, 1, 1, 114, 21, 31, 3, 6, 74, 63, 37, 37, 59, 62, and 37), and the applied method for data imputation included

- a constant value that has meaning within the domain, such as 0, distinct from all other values was imputed for categorical variables;
- a mean or median value for the column (feature) was imputed instead of the missing value, if the variable is continuous.

This leads to the fact that attributes such as presence of groups $-SO_3^-$/$-O-SO_3^-$, buffer, and cross-linking were filled in with 0 in missing values, indicating no presence of that group, effect, etc., and attribute values that are continuous, both mean and median estimation (mean/median of that column), were tested. The results when using Extra Tree Regressor with mean imputation are given in Table 3. The results when using median imputation were very similar to results with mean imputation and therefore were not presented. The results are reported based on 5-fold cross validation.

It can be seen that, when comparing results before outlier removal and data imputation and after these preprocessing steps, the results for the test set improved drastically, from an R^2 score on the test set of 0.92 before data imputation and outlier removal to an R^2 score of 0.98 after data imputation and outlier removal, with MAE of 116.06.

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

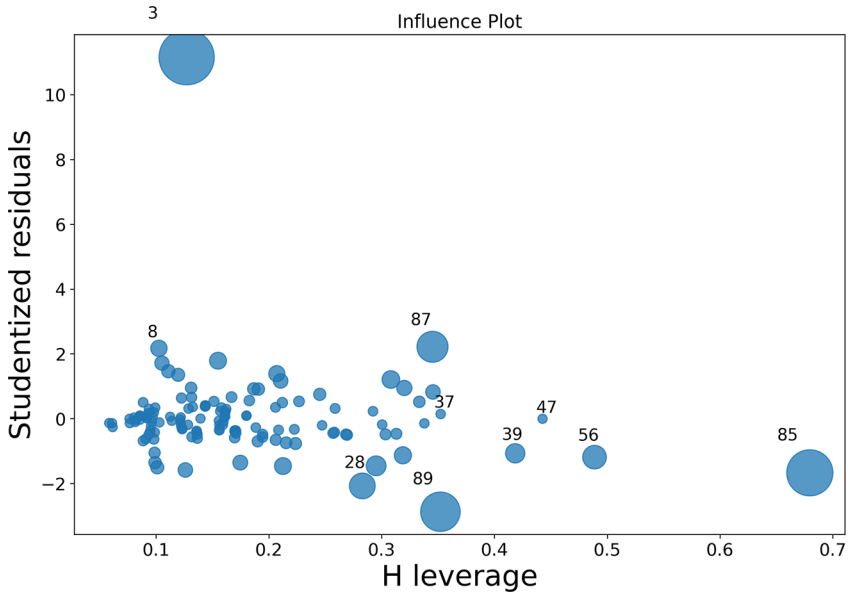


Figure 3. Leverage vs studentized residuals plot (original dataset).

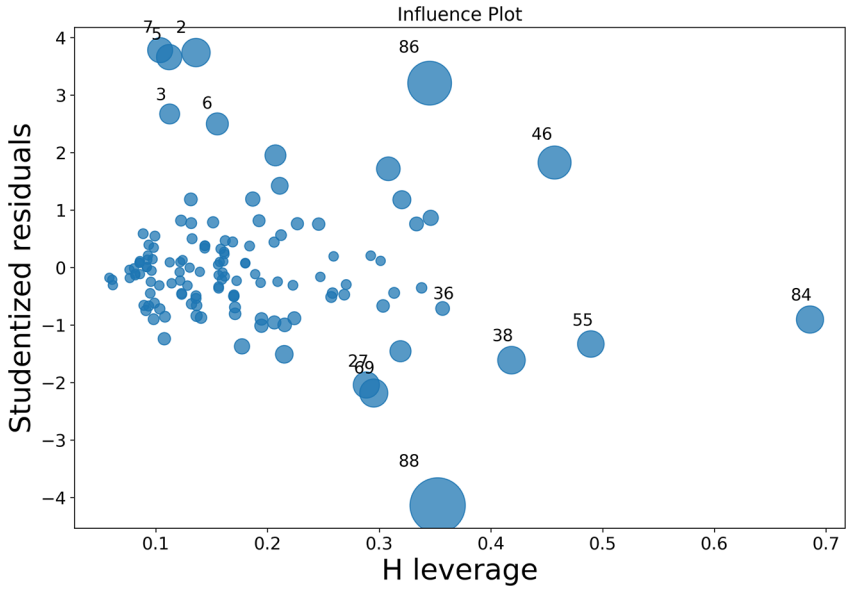


Figure 4. Leverage vs studentized residuals plot (dataset after removal of instance number 3).

Table 3. Results with Mean Imputation Using Extra Tree Regressor (after Outlier Removal)

Training subset	
R^2 score	1.000
MSE [nm]	4.13
RMSE [nm]	2.03
MAE [nm]	0.39
Test subset	
R^2 score	0.98
MSE [nm]	47633.85
RMSE [nm]	218.25
MAE [nm]	116.06

For hyperparameter optimization investigation via RandomizedSearchCV, a grid of hyperparameter ranges is defined and randomly sampled from the grid, performing K-Fold CV with each combination of values. This was done to avoid the overfitting problem, generally present in similar problems. Also, the obtained results show a large difference between the RMSE on a test subset in comparison to the training set, indicating overfitting. Therefore, we have implemented optimization of a number of hyperparameters such as number of trees in a forest (search space sequence of evenly spaced numbers in linear space with start = 95, stop = 111, and number of elements = 400), features to consider at every split ('auto' or 'sqrt'), maximum depth of the tree (search space sequence of evenly spaced numbers in linear space with start = 0, stop = 8, and number of elements = 100), minimum number of samples required to split a node ('none', 1,2,3,4,5,6,7), minimum number of samples required at each leaf node ('none', 1,2,3,4), and bootstrapping method of selecting samples for training each tree (true or false), which can be tuned to stop early the growth of the tree and prevent the model from overfitting. The best results were achieved with the

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

number of trees in a forest = 104, features to consider at every split = 'auto', maximum depth of the tree = 7, minimum number of samples required to split a node = 2, minimum number of samples required at each leaf node = 1, bootstrap = 'false'. The results are presented in Table 4.

Table 4. Results of Extra Tree Regressor with Default Hyperparameters, Optimized Hyperparameters (No Missing Data Imputation), and Optimized Hyperparameters (with Missing Data Imputation)

Extra Tree Regressor					
Default hyperparameters		Optimized hyperparameters: (no missing data imputation)		Optimized hyperparameters (with missing data imputation)	
Training					
R^2 score	1.00	R^2 score	0.99	R^2 score	0.999
MSE [nm]	4.13	MSE [nm]	42985.96	MSE [nm]	3532.13
RMSE [nm]	2.03	RMSE [nm]	207.33	RMSE [nm]	59.43
MAE [nm]	0.39	MAE [nm]	97.80	MAE [nm]	39.16
Test					
R^2 score	0.98	R^2 score	0.98	R^2 score	0.983
MSE [nm]	47633.85	MSE [nm]	52948.79	MSE [nm]	40598.73
RMSE [nm]	218.25	RMSE [nm]	230.10	RMSE [nm]	201.49
MAE [nm]	116.06	MAE [nm]	110.55	MAE [nm]	107.29

The results show that with optimal hyperparameters (with missing data imputation), the results are the best among all investigated solutions, which is expected. However, one of the drawbacks of the RandomizedSearchCV methodology is the defined search space for hyperparameters. There should be a balance between the search space and memory/time for optimization; as such, optimization relies on searching all the combinations in the defined search space. Should the space search be extended even more, better results may be achieved. Determining optimal hyperparameters in defined search space is a common challenge in ML.

Figure 5 presents the actual vs predicted values of coating thickness for the instances in the test set. It can be seen that even for larger values of coating thickness, the predicted values are close to actual values, which means that the model generalizes well to unseen data. It should be emphasized that the larger RMSE is due to the fact that there are also values of coating thickness, i.e., 5000−6000, meaning that these differences in predicted and actual values are going to increase the RMSE.

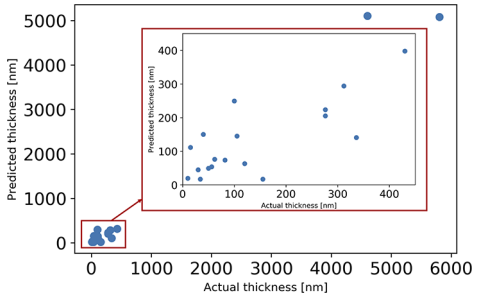


Figure 5. Predicted vs actual values of coating thickness for the original test dataset (zoomed in figure shows the values of thickness less than 450 nm).

To illustrate the point, if the test dataset was extracted in such a way that only instances with values for thickness are less than 100 nm, the results achieved have improved in terms of monitored statistics (MSE = 133.679, RMSE = 11.56, MAE = 8.06) (Figure 6). This means that some larger values of statistics are due to the dataset diversity and large differences in ranges in coating thickness.

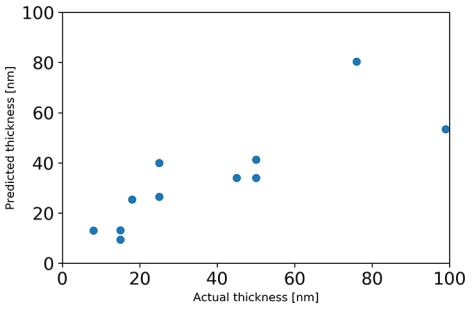


Figure 6. Predicted vs actual values of coating thickness for the test dataset with coating thickness less than 100 nm (model retrained when using only data where thickness is less than or equal to 100 nm).

In order to certify that there is no overfitting, we have introduced the SMOGN oversampling[41] and tested the same methodology on the nonskewed dataset. The results indicate that there is no overfitting, but rather larger values of RMSE are due to the nature of dataset. With the use of SMOGN oversampling, R^2 was 0.980, MSE = 46933.204, RMSE = 216.64, and MAE = 111.414 on the test dataset.

In addition, Select K best was used to determine 6 best predictors of coating thickness. By calculating the feature importance scores, it was determined that 6 predictors had higher scores than the rest of the attributes (by order of magnitude higher scores). A higher score means that the specific feature will have a larger effect on the model in predicting the target variable (coating thickness). As a result, these 6 predictors were adopted as the most influential predictors:

1. polyanion
2. number of bilayers
3. ending polymer
4. presence of negatively charged groups: COO^-
5. pH polycation
6. buffer concentration, mM

The results achieved with these 6 predictors on a whole dataset during the training phase were R^2 score = 0.989, MSE = 36225.257 nm, RMSE = 190.329 nm, MAE = 48.020 nm. This means that in the future, we can possibly use only these 6 predictors instead of 23 to accurately predict the coating thickness. Among these predictors, the number of bilayers is the most influential one; however, it also must be taken into account as the thickness generated by different polymers may vary significantly for the same number of bilayers as there are polymer couples that lead to a linear or exponential film growth regime.[42] Regarding other predictors, the type of polyanion and the presence of COO^- groups underline the importance of polyanion properties. This is associated to

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

polycation pH, which is also understandable, as the proper pH creates the optimal conditions for polyelectrolyte assembly and the level of assembly determines the coating thickness from the efficacy of the process. In its turn, buffer concentration should play on polymer conformation via ionic strength. The role of the ending polymer for LbL film thickness is less obvious. Even though, all these parameters are known for their effects on film formation, their relative importance has not been demonstrated in the literature and the proposed methodology enables us to see the parameters that contribute most to the final required properties and select the right zones of design space for optimized coatings.

Compared to the previous work on LbL coating thickness prediction,[33] here, we use almost two times more instances (131 vs 77) and do not need a complex step of molecular descriptor generation; only coating-related parameters are sufficient. Molecular descriptors provide a more in-depth understating of the underlying parameters, but some of the molecular descriptors are difficult to translate into actual physicochemical properties in an exploitable manner. Reliable models based on the weighted effect on controllable macroscale parameters will enable new facilitated functional coating design. In addition, the prediction looks more precise. This work shows that ML is a powerful method that allows accurate predictions using even small datasets containing missing values.

4. CONCLUSIONS

LbL film coatings are a powerful tool for surface modification, including in the biomedical field. However, their setup can take a long time and be very costly. ML methodologies that are now emerging can accelerate and improve new coating development and can potentially revolutionize the field.

ML-based methods have recently been used by researchers in many fields due to their special abilities in dealing with dynamic, nonlinear, categorical, and multidimensional regression and classification problems, where analytical solutions are difficult and time-consuming, if not impossible. As a result, these tools have received a lot of interest in the materials engineering community. ML algorithms demonstrated substantial benefits in reliably mapping polymer activity and resolving all forms of significant content and processing parameters. The current research has shown that ML algorithms are effective in predicting the coating output value, with the Extra Tree Regressor algorithm demonstrating superior predictive performance, when used in combination with optimized hyperparameters and with missing data imputation. The best predictors of the coating thickness are determined, and they can be later used to accurately predict the coating thickness and aid in the development of coatings with specific properties, avoiding the measurement of a large number of different parameters.

The main limitations of this study were that the literature dataset has many missing values, meaning that in the literature papers, they were not defined or reported. We have overcome this problem by using data imputation. Another limitation is that the range of the coating thickness in literature data was very wide, reflecting on the variance. However, the current results show that even with these limitations, we managed to set up good models for prediction, which can be further upgraded and extended to other similar problems.

As a continuation, the methodology can be adapted and used for predicting the outputs connected to the antimicrobial, anti-inflammatory, and antiviral properties in order to be able to respond to the actual biomedical problems such as antibiotic resistance, implant rejection, or COVID-19 outbreak.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsomega.2c06471.

Table S1: Original dataset and Table S2: Dataset with mean data imputation (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Nihal Engin Vrana − *SPARTHA Medical, Strasbourg 67100, France;* orcid.org/0000-0002-5398-6710; Email: evrana@sparthamedical.eu

Authors

Tijana Šušteršič − *Faculty of Engineering, University of Kragujevac (FINK), Kragujevac 34000, Serbia; Steinbeis Advanced Risk Technologies Institute doo Kragujevac (SARTIK), Kragujevac 34000, Serbia; Bioengineering Research and Development Center (BioIRC), Kragujevac 34000, Serbia*

Varvara Gribova − *Biomaterials and Bioengineering laboratory, INSERM UMR 1121, Strasbourg 67100, France; Université de Strasbourg, Faculté de Chirurgie Dentaire, Strasbourg 67000, France;* orcid.org/0000-0002-5175-2072

Milica Nikolic − *Steinbeis Advanced Risk Technologies Institute doo Kragujevac (SARTIK), Kragujevac 34000, Serbia; Institute of Information Technologies, University of Kragujevac, Kragujevac 34000, Serbia; Eindhoven University of Technology, Eindhoven 5611 CB, The Netherlands*

Philippe Lavalle − *Biomaterials and Bioengineering laboratory, INSERM UMR 1121, Strasbourg 67100, France; Université de Strasbourg, Faculté de Chirurgie Dentaire, Strasbourg 67000, France; SPARTHA Medical, Strasbourg 67100, France;* orcid.org/0000-0001-8798-912X

Nenad Filipovic − *Faculty of Engineering, University of Kragujevac (FINK), Kragujevac 34000, Serbia; Steinbeis Advanced Risk Technologies Institute doo Kragujevac (SARTIK), Kragujevac 34000, Serbia; Bioengineering Research and Development Center (BioIRC), Kragujevac 34000, Serbia*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.2c06471

Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Notes

The authors declare the following competing financial interest(s): Conflict of interest statement (NE Vrana, P Lavalle): SPARTHA Medical is a spin-off company from INSERM and this work has been carried out as a collaboration. SPARTHA Medical develops commercial coatings, but the current study does not focus on SPARTHA coatings and the aim was to set a new methodology for coating property prediction.

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

■ ACKNOWLEDGMENTS

This study was funded by the European Project H2020 PANBioRA [grant number 760921], from ANR TerminAnion, Bourse Frenchtech Emergence grant and Serbian Ministry of Education, Science, and Technological Development [451-03-68/2022-14/200107 (Faculty of Engineering, University of Kragujevac) and 451-03-68/2022-14/200378 (Institute for Information Technologies, University of Kragujevac)]. This article reflects only the author's view. The Commission is not responsible for any use that may be made of the information it contains.

■ REFERENCES

(1) Morais, J. M.; Papadimitrakopoulos, F.; Burgess, D. J. Biomaterials/Tissue Interactions: Possible Solutions to Overcome Foreign Body Response. *AAPS J.* **2010**, *12*, 188−196.
(2) VanEpps, J. S.; Younger, J. G. Implantable Device-Related Infection. *Shock* **2016**, *46*, 597−608.
(3) Lebaudy, E.; Fournel, S.; Lavalle, P.; Vrana, N. E.; Gribova, V. Recent Advances in Antiinflammatory Material Design. *Adv. Healthcare Mater.* **2021**, *10*, No. 2001373.
(4) Santos, M. R. E.; Fonseca, A. C.; Mendonça, P. V.; Branco, R.; Serra, A. C.; Morais, P. V.; Coelho, J. F. J. Recent Developments in Antimicrobial Polymers: A Review. *Materials* **2016**, *9*, 599.
(5) Boudou, T.; Crouzier, T.; Ren, K.; Blin, G.; Picart, C. Multiple Functionalities of Polyelectrolyte Multilayer Films: New Biomedical Applications. *Adv. Mater.* **2010**, *22*, 441−467.
(6) Pahal, S.; Gakhar, R.; Raichur, A. M.; Varma, M. M. Polyelectrolyte multilayers for bio-applications: recent advancements. *IET Nanobiotechnol.* **2017**, *11*, 903−908.
(7) Mutschler, A.; Tallet, L.; Rabineau, M.; Dollinger, C.; Metz-Boutigue, M.-H.; Schneider, F.; Senger, B.; Vrana, N. E.; Schaaf, P.; Lavalle, P. Unexpected Bactericidal Activity of Poly(arginine)/Hyaluronan Nanolayered Coatings. *Chem. Mater.* **2016**, *28*, 8700−8709.
(8) Özçelik, H.; Vrana, N. E.; Gudima, A.; Riabov, V.; Gratchev, A.; Haikel, Y.; Metz-Boutigue, M.-H.; Carradò, A.; Faerber, J.; Roland, T.; Klüter, H.; Kzhyshkowska, J.; Schaaf, P.; Lavalle, P. Harnessing the Multifunctionality in Nature: A Bioactive Agent Release System with Self-Antimicrobial and Immunomodulatory Properties. *Adv. Healthcare Mater.* **2015**, *4*, 2026−2036.
(9) Mutschler, A.; Betscha, C.; Ball, V.; Senger, B.; Vrana, N. E.; Boulmedais, F.; Schroder, A.; Schaaf, P.; Lavalle, P. Nature of the Polyanion Governs the Antimicrobial Properties of Poly(arginine)/Polyanion Multilayer Films. *Chem. Mater.* **2017**, *29*, 3195−3201.
(10) Shah, N. J.; Macdonald, M. L.; Beben, Y. M.; Padera, R. F.; Samuel, R. E.; Hammond, P. T. Tunable dual growth factor delivery from polyelectrolyte multilayer films. *Biomaterials* **2011**, *32*, 6183−6193.
(11) Bouyer, M.; Guillot, R.; Lavaud, J.; Plettinx, C.; Olivier, C.; Curry, V.; Boutonnat, J.; Coll, J.-L.; Peyrin, F.; Josserand, V.; Bettega, G.; Picart, C. Surface delivery of tunable doses of BMP-2 from an adaptable polymeric scaffold induces volumetric bone regeneration. *Biomaterials* **2016**, *104*, 168−181.
(12) Boudou, T.; Kharkar, P.; Jing, J.; Guillot, R.; Pignot-Paintrand, I.; Auzely-Velty, R.; Picart, C. Polyelectrolyte multilayer nanoshells with hydrophobic nanodomains for delivery of Paclitaxel. *J. Controlled Release* **2012**, *159*, 403−412.
(13) Hu, B.; Chen, L.; Lan, S.; Ren, P.; Wu, S.; Liu, X.; Shi, X.; Li, H.; Du, Y.; Ding, F. Layer-by-Layer Assembly of Polysaccharide Films with Self-Healing and Antifogging Properties for Food Packaging Applications. *ACS Appl. Nano Mater.* **2018**, *1*, 3733−3740.
(14) Rivero, P.; Goicoechea, J.; Arregui, F. Layer-by-Layer Nanoassembly: A Powerful Tool for Optical Fiber Sensing Applications. *Sensors* **2019**, *19*, 683.
(15) Lutkenhaus, J. L.; Hammond, P. T. Electrochemically enabled polyelectrolyte multilayer devices: from fuel cells to sensors. *Soft Matter* **2007**, *3*, 804−816.
(16) Xiao, F.-X.; Pagliaro, M.; Xu, Y.-J.; Liu, B. Layer-by-layer assembly of versatile nanoarchitectures with diverse dimensionality: a new perspective for rational construction of multilayer assemblies. *Chem. Soc. Rev.* **2016**, *45*, 3088−3121.
(17) Ghaffari, A.; Abdollahi, H.; Khoshayand, M. R.; Bozchalooi, I. S.; Dadgar, A.; Rafiee-Tehrani, M. Performance comparison of neural network training algorithms in modeling of bimodal drug delivery. *Int. J. Pharm.* **2006**, *327*, 126−138.
(18) Ghazanfari, N.; Gholami, S.; Emad, A.; Shekarchi, M. Evaluation of GMDH and MLP Networks for Prediction of Compressive Strength and Workability of Concrete. *Bull. Soc. R. Sci. Liege* **2017**, *86*, 855−868.
(19) Chen, L.; Raman, K. A Subjective Design Framework for Conceptual Design of Polymeric Processes with Multiple Parameters. *Res. Eng. Des.* **2000**, *12*, 220−234.
(20) Vafaeenezhad, H.; Asadolahpour, S. R.; Nayebpashaee, N.; Seyedein, S. H.; Aboutalebi, M. R. Intelligent use of data to optimize compressive strength of cellulose-derived composites. *Appl. Soft Comput.* **2016**, *40*, 594−602.
(21) Wu, D.; Jennings, C.; Terpenny, J.; Gao, R. X.; Kumara, S. A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests. *J. Manuf. Sci. Eng.* **2017**, *139*, No. 071018.
(22) Kotsiantis, S. B.; Zaharakis, I.; Pintelas, P. Supervised machine learning: A review of classification techniques Emerging Artificial Intelligence. *Appl. Comput. Eng.* **2007**, *160*, 3−24.
(23) Nikam, S. A comparative study of classification techniques in data mining algorithms. *Orient. J. Comput. Sci. Technol.* **2015**, *8*, 13−19.
(24) Soundarya, M.; Balakrishnan, R. Survey on classification techniques in data mining. *Int. j. innov. sci. eng. technol.* **2014**, *3*, 7550−7552.
(25) Wuest, T.; Weimer, D.; Irgens, C.; Thoben, K.-D. Machine learning in manufacturing: advantages, challenges, and applications. *Prod. Manuf. Res.* **2016**, *4*, 23−45.
(26) Dey, A. Machine Learning Algorithms: A Review. *Comput. Sci* **2016**, *9*, 381−386.
(27) Altarazi, S.; Ammouri, M.; Hijazi, A. Artificial neural network modeling to evaluate polyvinylchloride composites' properties. *Comput. Mater. Sci.* **2018**, *153*, 1−9.
(28) Mallakpour, S.; Hatami, M.; Khooshechin, S.; Golmohammadi, H. Evaluations of thermal decomposition properties for optically active polymers based on support vector machine. *J. Therm. Anal. Calorim.* **2014**, *116*, 989−1000.
(29) Wei, Q.; Melko, R. G.; Chen, J. Z. Y. Identifying polymer states by machine learning. *Phys. Rev. E* **2017**, *95*, No. 032504.
(30) Wu, X.; Zhang, H.; Cui, H.; Ma, Z.; Song, W.; Yang, W.; Jia, L.; Zhang, H. Quantitative Relationship Analysis of Mechanical Properties with Mg Content and Heat Treatment Parameters in Al¯7Si Alloys Using Artificial Neural Network. *Materials* **2019**, *12*, 718.
(31) Altarazi, S.; Allaf, R.; Alhindawi, F. Machine Learning Models for Predicting and Classifying the Tensile Strength of Polymeric Films Fabricated via Different Production Processes. *Materials* **2019**, *12*, 1475.
(32) Wang, H.; Hsieh, S.-J.; Peng, B.; Zhou, X. Non-metallic coating thickness prediction using artificial neural network and support vector machine with time resolved thermography. *Infrared Phys. Technol.* **2016**, *77*, 316−324.
(33) Gribova, V.; Navalikhina, A.; Lysenko, O.; Calligaro, C.; Lebaudy, E.; Deiber, L.; Senger, B.; Lavalle, P.; Vrana, N. E. Prediction of coating thickness for polyelectrolyte multilayers via machine learning. *Sci. Rep.* **2021**, *11*, 18702.
(34) Agarwal, A.Linear Regression using Python. *Towards Data Science***2018**, https://towardsdatascience.com/linear-regression-using-python-b136c91bf0a2, (accessed November 1, 2020).

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686

(35) Sharp, T.An Introduction to Support Vector Regression (SVR). *Towards Data Science***2020**https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2, (accessed November 5, 2020).
(36) Malik, U.Random Forest Algorithm with Python and Scikit-Learn. *Stack Abuse***2018**, https://stackabuse.com/random-forest-algorithm-with-python-and-scikit-learn/, (accessed November 12, 2020).
(37) Brownlee, J.How to Develop an Extra Trees Ensemble with Python. *Machine Learning Mastery***2021**, https://machinelearningmastery.com/extra-trees-ensemble-with-python/, (accessed November 2, 2020).
(38) Koehrsen, W.Hyperparameter Tuning the Random Forest in Python. *Towards Data Science***2018**, https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74, (accessed December 15, 2020).
(39) sklearn.feature_selection.SelectKBest. *Scikit-learn*; Packt Publishinghttps://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html (accessed December 5, 2020).
(40) Colin Cameron, A.; Windmeijer, F. A. G. An R-squared measure of goodness of fit for some common nonlinear regression models. *J. Econom.* **1997**, *77*, 329−342.
(41) Branco, P.; Torgo, L.; Ribeiro, R. P., SMOGN: a Pre-processing Approach for Imbalanced Regression. In *Proceedings of the First International Workshop on Learning with Imbalanced Domains: Theory and Applications*, Paula Branco Luís, T.; Nuno, M., Eds. Proc. Mach. Learn. Res.2017, *74*, 36−50.
(42) Picart, C.; Mutterer, J.; Richert, L.; Luo, Y.; Prestwich, G. D.; Schaaf, P.; Voegel, J.-C.; Lavalle, P. Molecular basis for the explanation of the exponential growth of polyelectrolyte multilayers. *Proc. Natl. Acad. Sci. U. S. A.* **2002**, *99*, 12531−12535.

https://doi.org/10.1021/acsomega.2c06471
ACS Omega 2023, 8, 4677−4686