Improving Chemical Biodegradability Prediction Using Logistic Regression and QSAR Data: Stratified Cross Validation Approach

***Abstract*—This report goes into detail explaining a machine learning approach to predicting the biodegradable nature of given chemicals using the Quantitative Structure-Activity Relationships (QSAR) data set provided. The dataset consists of a row of features for the list of chemicals mentioned, with a binary label at the end of each row to inform the user whether the chemical is biodegradable or not. A logistic regression model is used to classify chemicals according to their 41 characteristics. To make sure the results will be generalized and consistent when introduced to chemicals not in the dataset, a 5-fold cross-validation is used, while also focusing on metrics such as accuracy, precision, recall, F1-score, and Area Under the Curve (AUC) to aid in analyzing the performance of the model. Finally, the model is trained on the entire dataset, to obtain a high accuracy and AUC score, and to also make the model effective in distinguishing between the biodegradable and non-biodegradable chemicals. In general, the results show this algorithm provides a reliable solution to predicting the biodegradable nature of these chemicals.**

I. Introduction

Predicting the chemical biodegradability is a necessary step in analyzing the environmental effects it can have on the ecosystem, In addition, accumulation of these chemical substances could be hazardous or toxic due to some of its characteristics, and could be harmful to its surroundings when its either produced, disposed of as a waste or a by-product of some industrial processes. The accumulation will result in increased exposure and chemical concentrations causing significant health and ecological risks over long periods of time, which makes it necessary to predict the biodegradability in advance [1]. These chemical could also severely affect the biome of that environment, by entering the organisms through food or water, and thereby disrupting the biological life-cycle like long term effects in the food chain of plants and animals.

According to the REACH framework a regulating commission, chemicals that are produced need to undergo a biodegradation test, but its challenging to test all the chemical substances in the market. Chemicals are produced in tons per year,however only 61 percent have substance characteristics concerning its biodegradability available to analyze[1]. Due to the above mentioned challenges, the QSAR (Quantitative Structure-Activity Relationships) technique aims to predict the biological or chemical activity based on the molecular structure[1]. In this report machine learning techniques have been applied to logistic regression model to reliably predict the biodegradability of 1055 chemical substances using the QSAR data. This model is reliant on a 2 class classification solution as the labels are of binary output. To reduce overfitting of the model, feature selection is introduced by implementing L1 regularization (Lasso), building on the works of Tibshirani[2].This technique helps to identify certain molecular descriptors necessary for prediction, while shrinking the other feature coefficients to zero.

To make sure that the model is robust and also responsive to other chemicals that follow the same 41 features in the dataset, a 5-fold stratified cross-validation method is implemented, to make sure its provides reliable and generalized results to data outside initial datasets provided[3]. In addition to generalizing models, its also required to standardize the molecular features of the dataset provided. This can be done by normalizing the data using the Z-score technique, which ensures that the some molecular descriptors that are the features in this dataset, do not dominate the other features due to the differences in scale. The performance of the model was evaluated using various parameters like accuracy, precision, recall, F1-score, and ROC characteristic plot with AUC [4][5][6]. This study builds on the previous work done in machine learning for biodegradability prediction, aiming to provide a reliable algorithm for improving the overall efforts in taking preventive measures to limit the use and spread of chemicals that are not degradable in nature, and thereby encourage development of alternative, safe, and environment friendly chemical products.

II. Data Processing

The QSAR dataset [1] implemented to train the logistic regression model, consists of 1,055 chemicals, each described by 41 molecular features. The target variable is a binary label indicating whether a chemical is biodegradable (1) or non-biodegradable (0). To make sure that the model is trained on consistent, reliable and well prepared data, the QSAR dataset is first checked for any missing values which could produce biased data, referred from the machine learning repository co-authored by Mansouri [7].

The dataset is first split into the X and Y matrix, for ease of handling features and target value data separately. The X matrix which includes the features, are standardized by implementing Z-score Normalization technique, which ensures that all the features are of similar scale. It adjusts the data so that the feature data in each columns are scaled to have a mean of 0 and standard deviation of 1.

The Z-score normalization equation is given by:

$$Z = \frac{X - \mu}{\sigma} \tag{1}$$

where:

- Z is the Z-score or normalized value,
- X is the original value from the dataset,
- μ is the mean of the dataset,
- σ is the standard deviation of the dataset.

Using the above equation X matrix gets replaced with a matrix of same dimensions where each element is normalized. This step is critical for regression algorithm, where large range features could heavily influence the model's weight coefficients. The Z-score normalization works really well with datasets having features of varying magnitudes just like the datasets being used for this model.

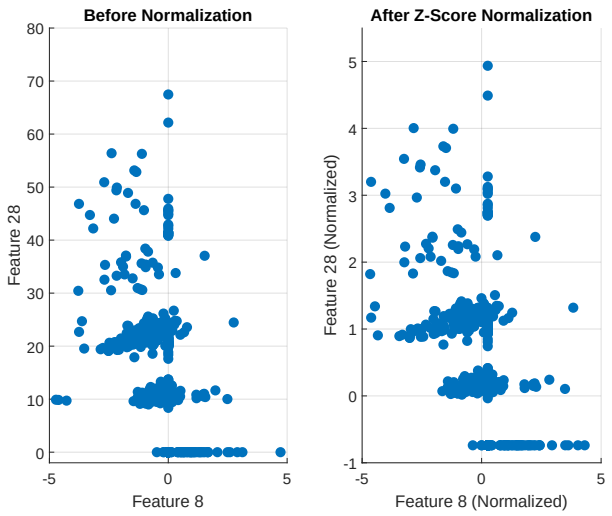

Fig. 1. Scatter plot showcasing the Z-score Normalization technique to rescale the features in the dataset.

In Figure 1 the dataset's improvement in scale is noticeable by plotting a comparative plot between two different features from the dataset. Prior to normalization the scatter was skewed towards the y axis, due to the influence of feature 28, going past 60 units in terms of magnitude. After normalization feature 28 has been scaled down within minimum and maximum range of (-1,5) respectively.

A bias term was finally added to the normalized feature matrix to make sure that the model could make predictions on the complete dataset, which increases it's flexibility in adjusting the baseline probability for class identification. In simpler terms, the decision boundary which segregates the classes, is able to move freely towards and away from the data points.

III. Methodology

This section outlines the working of the primary decision model used, and the various performance optimization methods implemented to reduce over-fitting and improve the learning rate of the model.

A. Logistic Regression -

Logistic regression, as a supervised learning model, is quite simple and effective in solving 2-class classification problems. Compared to a neural network, where the linear, Re-Lu, or Sigmoid function could be any activation function for each of its 'neurons', and would hence make the underlying operation of the system less interpretable, logistic regression is more interpretable because it directly models the relationship between input features and the probability of the outcome, with each feature's effect clearly represented by its respective coefficient.

The logistic regression function for multiple features is given by:

$$P(Y = 1 \mid X) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n)}} \tag{2}$$

Alternatively, the vectorized form is:

$$P(Y = 1 \mid X) = \frac{1}{1 + e^{-(\theta^T X)}} \tag{3}$$

where:

- $P(Y = 1 \mid X)$ is the probability of Y being 1 given the input features $X = (X_1, X_2, \ldots, X_n)$.
- θ_0 is the intercept term (bias).
- $\theta_1, \theta_2, \ldots, \theta_n$ are the weights corresponding to the features $X_1, X_2, \ldots, X_n$.
- $X_1, X_2, \ldots, X_n$ are the input features.
- e is the exponential term.

Equation (3) is a sigmoid function which transforms the input linear equation $(\theta_0 + \theta_1 X_1 + \theta_2 X_2 + \cdots + \theta_n X_n)$ into probabilistic outputs between ranges of 0 and 1.

B. L1 Regularization -

To prevent over-fitting and implement feature selection of the model, the L1 (Lasso) Regularization is used [2]. L1 regularization like other regularization techniques adds penalty to the overall Loss Function. The advantage of implementing L1 is that it helps in shrinking some coefficients of less important features to zero. This makes the model sparse, where only the relevant features remain, thereby enhancing feature selection.

The regularized cost function for logistic regression is given by:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(h_\theta(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)})) \right] + \frac{\lambda}{2} \sum_{j=1}^{n} |\theta_j| \tag{4}$$

where:

- $J(\theta)$ is the regularized cost function,
- m is the number of training examples,
- $y^{(i)}$ is the true label for the i-th training example,
- $h_\theta(x^{(i)}) =$ is the sigmoid function,
- $\frac{\lambda}{2} \sum_{j=1}^{n} |\theta_j|$ is the regularization term,
- θ_j is the model parameter for the j-th feature.

C. Stratified Cross Validation -

In stratified 5-fold cross-validation, data is split into five equal subsets. To make sure these folds are balanced, the algorithm takes in equal amounts of opposing classes, to maintain the similar class distribution in each fold. In this case,

the number of biodegradable and non-biodegradable chemicals as identified by their target values are of the same proportion in each set of test set in every fold. At the end of every fold after finding out the weight coefficients with the test set, the weight coefficients are implemented on the validation set and performance is noted.

D. Iterative Update of weights using Newton's Method -

Newtons method is used for optimizing and updating the models parameters through faster descent in approaching the optimal coefficients. Newton's method is implemented to iteratively update the θ weights by minimizing the Loss Function. Newton's method uses second-order derivatives also know as the Hessian Matrix, to make more accurate updates in each iteration. Faster convergence is achieved with this method.

The weight update rule for Newton's method is given by:

$$\theta = \theta - H^{-1}\nabla J(\theta) \tag{5}$$

Where:

- θ is the parameter vector (weights) that we are updating. These represent the coefficients in the model that we are optimizing.
- H is the Hessian matrix, which is the matrix of second-order partial derivatives of the cost function $J(\theta)$.
- $\nabla J(\theta)$ is the gradient (first derivative) of the cost function $J(\theta)$ with respect to the parameters θ. It indicates the direction in which the cost function is increasing most rapidly.
- The update rule $\theta = \theta - H^{-1}\nabla J(\theta)$ subtracts the product of the inverse Hessian and the gradient from the current parameter values.

IV. Model Analysis

The performance of the Logistic Regression model with Regularization was during the fold iterations of 5-fold Cross Validation stage and after the model was trained withe entire dataset to showcase the comparison of the performance of the model with incomplete data and response of the model with the new data which is not part of the initial dataset.This section analyses the model's performance using different performance metrics, including Accuracy, Precision, Recall, Specificity, F1-score, ROC Curve, and AUC. The goal is for the model to generalize to new data while also maintaining a level of accuracy.

To ensure that the model is performing well in terms of fitting to data not not present in the test set, the evaluation of the accuracy, specificty, precision, F1-score, ROC curve and the AUC is plotted and noted down at every fold iteration to observe the model prediction accuracy when dealing with inputs from the validation sets.

In Figure 2 it clearly shows the high performance levels of the model that is trained. The mean performance plots are very close to the final complete dataset trained model, which shows that the model is accurate for new sets of data that it hasn't been trained upon. A slight improvement is observed with the complete dataset trained model, due to further data received.

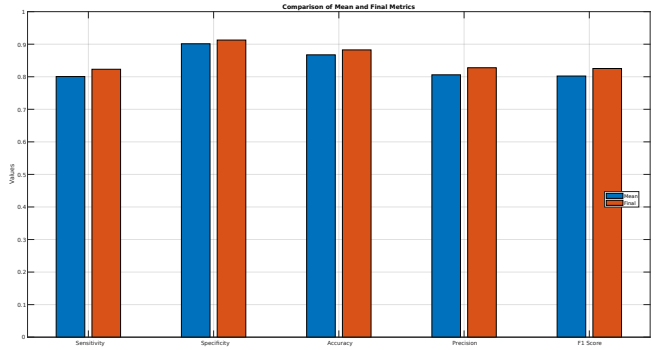

Fig. 2. Double bar plot showcasing the comparison of mean performance parameters and its final parameter after complete dataset training

Going into detail, we observe -

- Mean Accuracy : 0.8673
- Mean Precision : 0.81029
- Mean Recall(Sensitivity) : 0.7491
- Mean Specificity : 0.90414
- Mean F1 score : 0.80165
- Final Accuracy : 0.88246
- Final Precision : 0.82768
- Final Recall(Sensitivity) : 0.82303
- Final Specificity : 0.91273
- Final F1 score : 0.82535

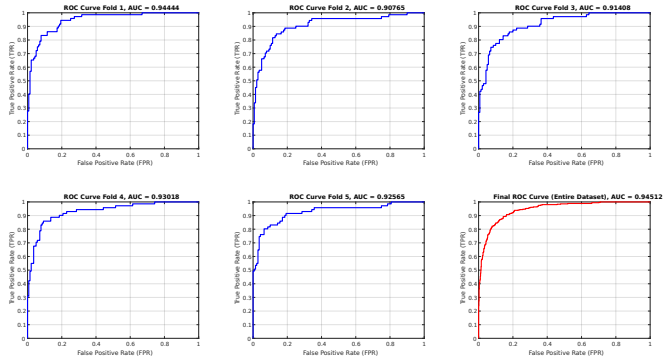

Fig. 3. ROC curve plot with respective AUC value of fold iterations and the final

According to Figure 3, we observe that the average of the AUC at every K-fold iteration is above 0.9 which is quite a high score. The same can be said for the final AUC value. Thesae results indicate that the model performed well, with high precision and recall, and minimized false positives and negatives.

as seen in Figure 4 the confusion matrix maintained a high true postive rate while also keeping the false positive rates relatively low. It identifies the chemicals with minimal errors.It should also be noted that the top left and bottom right quadrants are extremely high compared to the bottom left and top right quadrants with the later not crossing 100.

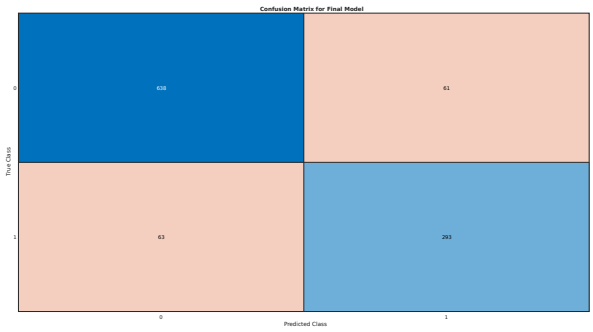

Fig. 4. ROC curve plot with respective AUC value of fold iterations and the final

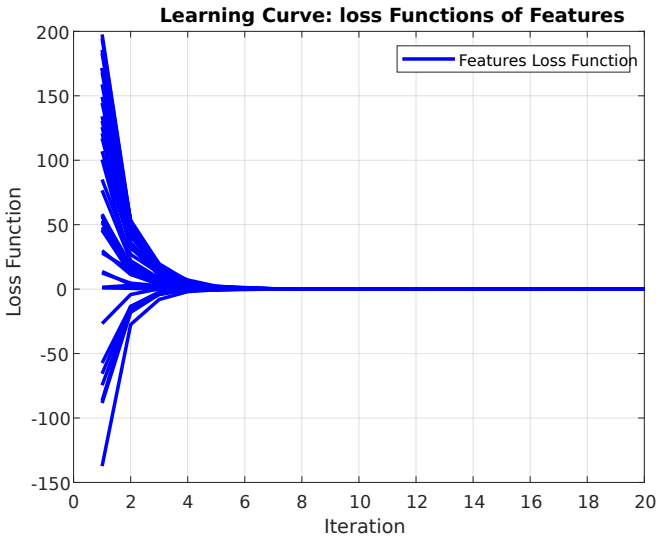

Fig. 5. ROC curve plot with respective AUC value of fold iterations and the final

As seen in Figure 5 there is a sharp decline in the loss functions of the 41 features near the 2nd iteration, and finally reaches nearly zero at 6 iterations. the model is shown for 20 iterations to show the model efficiency in reducing errors.

V. Conclusions and Recommendations

In this study, a logistic regression model with L1 regularization was successfully applied to the task of predicting the biodegradability of chemicals based on QSAR data. The model achieved strong performance across all evaluation metrics, including a high AUC score of 0.94512 on the entire dataset. The application of L1 regularization not only helped prevent over-fitting but also resulted in a more interpretable model by selecting only the most relevant features. Cross-validation confirmed the model's generalizability, with consistent performance across different data folds.

While the logistic regression model performed well, there are several opportunities for future improvement. Exploring more complex machine learning models, such as Random Forests or Support Vector Machines, may yield even better performance. Additionally, further hyper-parameter tuning and feature engineering could improve the overal model's predictive accuracy. However, the current model provides a reliable and interpretable solution for predicting the biodegradability of chemicals, making it a valuable tool for environmental protection efforts.

References

[1] K. Mansouri, T. Ringsted, D. Ballabio, R. Todeschini, and V. Consonni, "Quantitative structure–activity relationship models for ready biodegradability of chemicals," *Journal of Chemical Information and Modeling*, vol. 53, no. 4, pp. 867–878, 2013.

[2] Tibshirani, R., "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[3] Kohavi, R., "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol. 2*, 1995.

[4] Jin Huang and C. X. Ling, "Using AUC and accuracy in evaluating learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 299-310, Mar. 2005, doi: 10.1109/TKDE.2005.50.

[5] C.J. van Rijsbergen, "Foundations of evaluation," *Journal of Documentation*, vol. 30, pp. 365-373, 1974.

[6] Y. Sasaki, "The truth of the F-measure," *Teach. Tutor. Mater.*, 2007.

[7] UCI Machine Learning Repository, "QSAR Biodegradation," [Online]. Available: https://archive.ics.uci.edu/dataset/254/qsar+biodegradation. [Accessed: Dec. 18, 2024].