# Automated Machine Learning Tool: The First Stop for Data Science and Statistical Model Building

DeepaRani Gopagoni[1], P V Lakshmi[2]
Department of Computer Science and Engineering
GIT GITAM (Deemed to be University)
Vishakhapatnam, Andhra Pradesh, India

*Abstract*—**Machine learning techniques are designed to derive knowledge out of existing data. Increased computational power, use of natural language processing, image processing methods made easy creation of rich data. Good domain knowledge is required to build useful models. Uncertainty remains around choosing the right sample data, variables reduction and selection of statistical algorithm. A suitable statistical method coupled with explaining variables is critical for model building and analysis. There are multiple choices around each parameter. An automated system which could help the scientists to select an appropriate data set coupled with learning algorithm will be very useful. A freely available web-based platform, named automated machine learning tool (AMLT), is developed in this study. AMLT will automate the entire model building process. AMLT is equipped with all most commonly used variable selection methods, statistical methods both for supervised and unsupervised learning. AMLT can also do the clustering. AMLT uses statistical principles like R2 to rank the models and automatic test set validation. Tool is validated for connectivity and capability by reproducing two published works.**

***Keywords—Automated machine learning; regression models; support vector machines; QSAR; QSPR; artificial neural networks; k-means clustering; R program; shiny web app; drug design; market analysis; supervised learning; Naive Bayes classification***

## I. Introduction

In drug discovery and environmental toxicology, QSAR models regarded as a scientifically credible tool for predicting and classifying the biological activities of untested chemicals[1]. There is a lot of correlation and overlap between QSAR methods and general machine learning methods. In both QSAR and machine learning (ML) methods, descriptors have derived from factors that can affect the response variable. ML techniques designed to derive knowledge out of existing data on the fundamental of “Stored data becomes useful only when it has analyzed and turned into information that can make use of, for example, to make predictions” [2-4]. Many studies are available highlighting a successful application of ML techniques to diverse problems, which ranges from pharmaceutical industries to environmental sciences, mobile viruses, e-commerce, and business problems [5-7]. Machine learning methods have multiple applications like Genomic Medicine, Econometric Approach, Manufacturing, Solar radiation forecasting, big data learning, discovering phase transitions and Banking industry. Training set selection coupled with variable reduction and selection of statistical methods will make the number of models and combinations are high in number[8, 9]. Thus takes up a lot of time for model building and analysis followed by rebuilding the model with the different training set. Therefore having the tool which makes automatic data processing, training set selection, dimensional reduction, model building using multiple machine learning algorithms and model reporting will significantly save time by removing the multiple hit and trial approach. Thus the practitioner/Data scientist can focus on analysis for the progress of the project. AMLT provides users an integrated and friendly tool to build all machine learning models at one place. AMLT is freely available at "https://automatedmachine learning-gitamcse.shinyapps.io/MLPv3/"

One of the attempts made to automate the model building is in microsimulation. Modgen uses a single Montecarlo-based method to automate the micro simulation model building for different new data types, another attempt is AutoML, and automated prediction of the enzymatic functions of uncharacterized proteins, is an important topic in the field of bioinformatics. Both tools have focused only on single statistical method[10, 11].

Some challenges that data-set can contain are, e.g. missing values, high-dimensional data, mixing of numerical and text variables, non standard data, irrelevant and redundant information which may impact the performance of learning algorithms[12]. Today, most machine learning techniques handle only data with continuous and nominal values[12,13]. Missing values issue represents a very common challenge, there is a large amount of literature and practical solutions (e.g. in R) available[14, 15, 16, 17]. Pre-processing of data has a critical impact on the results. This can present challenges for the training of certain algorithms.

Proposed tool will enable practitioners to focus on the collection of good data sets and analyze the models for productivity. The tool will automate the model building process by picking the training set, variables and statistical methods that suits best to the input dataset.

### A. Defining the Problem Statement

Performance of each algorithm depends on the data available, data pre-processing and parameter settings. The best fitting algorithm has to be found by testing various ones in a realistic data.

As of today, the generally accepted approach to select a suitable ML algorithm for a certain problem is as follows:

First, one looks at the available data and how it is described (labeled, unlabeled, available expert knowledge, etc.) to choose between a supervised, unsupervised approach.

Secondly, the general applicability of available algorithms with regard to the research problem requirements (e.g. able to handle high dimensionality) has to be analyzed. A specific focus has to be laid on the structure, the data types and overall amount of the available data, which can be used for training and evaluation.

Thirdly, previous applications of the algorithms on similar problems are to be investigated in order to identify a suitable algorithm. The term 'similar' in this case, research problems with comparable requirements e.g. in other disciplines or domains.

Another challenge is the interpretation of the results. It has to be taken into account that not only the format or illustration of the output is relevant for the interpretation but also the specifications of the chosen algorithm. Within the interpretation of the results, certain more distinct limitations (again depending on the chosen algorithm) can have a large impact. Among those are, e.g. immune to over-fitting, bias, and variance (therefore bias-variance tradeoff)[18, 19].

However, one of the promising approach to select a suitable algorithm is to look for similar and analyze what ML algorithm was used to solve it and interpretation of results. Once the algorithm is applied, based on first results different methods can be applied to improve the model. However, this is very time consuming and iterative process to compare the models for selecting the best data, right algorithm, and ease of model interpretation. Therefore, this project is proposed on automation tool which considers the good practices of machine learning methods to build best predictive model suites to the problem. Fig. 1 explains the machine learning model building.

## II. Methodology

The program is written in blocks to incorporate data processing, supervised learning and unsupervised learning. Individual steps are described below.

### A. Building an R program

R began in the early 1990s as the personal project of Ross Ihaka and Robert Gentleman, R is the most popular open source statistical software. R and its add-on packages provide a wide range of options for data processing, statistical methods, and high performance computing. R program, which can transform the data in a flow is mentioned in Fig. 1 will be highly useful. The program could enable practitioners to focus on the collection of data sets, descriptors calculation and analyzing the models for productivity. Automatic tool will not contain any new or modified algorithms intentionally; it uses the collection of published algorithms. That makes tool can be benchmarked against known data sets and users can easily jump on to tool to start working without having the questions related validation of algorithms.

### B. Implementation and Features in the Tool

Coding the program is step by step process. Each step has few key feature implementations. Total code can be split into five main sections viz. Exploratory Data analysis (EDA), Data set selection, Feature selection, Selection of statistical method and results visualization. More details are given below.

*1) Exploratory Data analysis*: Returns the data header. Displays the column headings in data tables. Describes basic statistics of data i.e. summary (df), Plot Response variable/Output data to visualize distribution i.e. scatter or smooth plot.

*2) Data set selection*: Automatic random splitting of data into training and test sets. This splitting can be doable at different compositions of training and test tests.

*3) Feature selection*: Dimensionality reduction is a common technique used to reduce the number of variables in Machine Learning. The tool is equipped with multiple feature selection methods.

*4) Statistical method*: Most of the available statistical methods are coded together in this important section. All methods coded in a way that they process the same dataset for model building and test set validation. Parsing the single data set at onetime give the user to select the right algorithm for chosen data and set of variables. Both training and Test set validation is also automated to enable the user to choose the right model. This section divided into Classification, Regression and Clustering techniques. In classification, it supports (K-NN, Random Forest, Naive Bayes, SVM, ANN). In Regression, it supports (Linear Regression, Logistic Regression, Random Forest Regression, PCR, and PLSR). In clustering, tool supports K-MEANS Clustering

*5) Results visualization*: It's always very important to visualize the results in a manner results can be compared and interpreted. This is more important especially when multiple models are generated for training set and test set validation happens automatically.

Results visualization includes, For Classification TP= True Positive, FP=False Positive, TN=True Negative, FN=False Negative, Q=Q-Value, SE=Sensitivity. For Regression R2=R-Squared Value, RMSE=Root Mean Squared Error.

*6) Export results:* All algorithm results are exported at once to .csv file with predicted output added with testing data. Fig. 2 depicts the flow of the data how it is implemented.

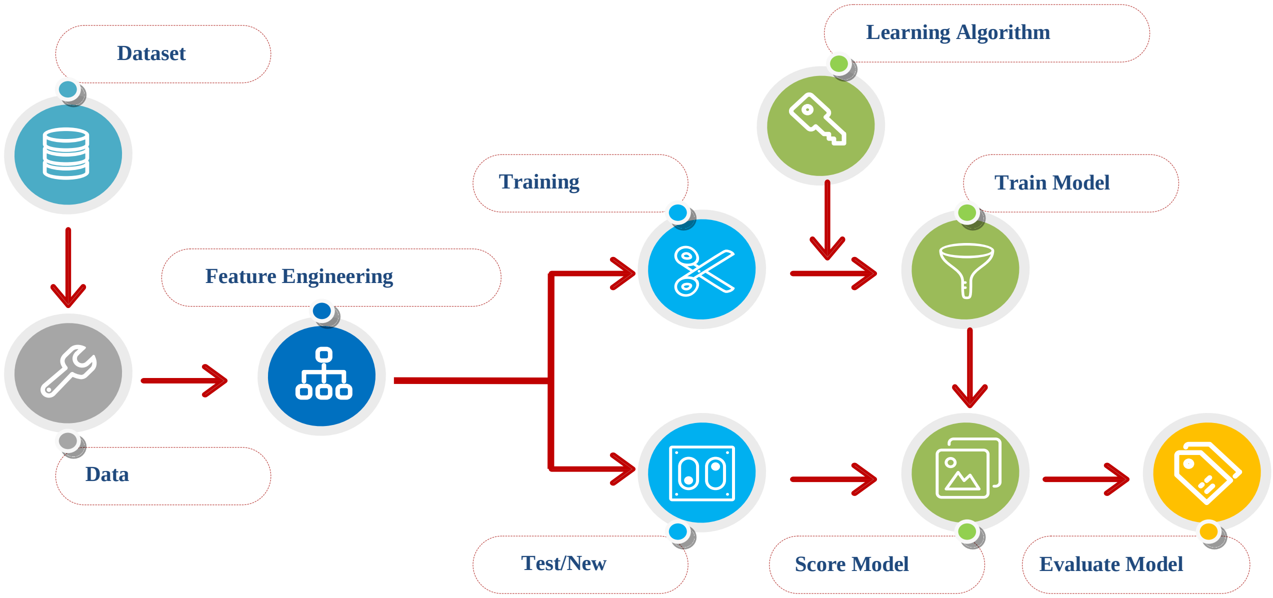


Fig. 1. Example flow for Machine Learning Model Building.

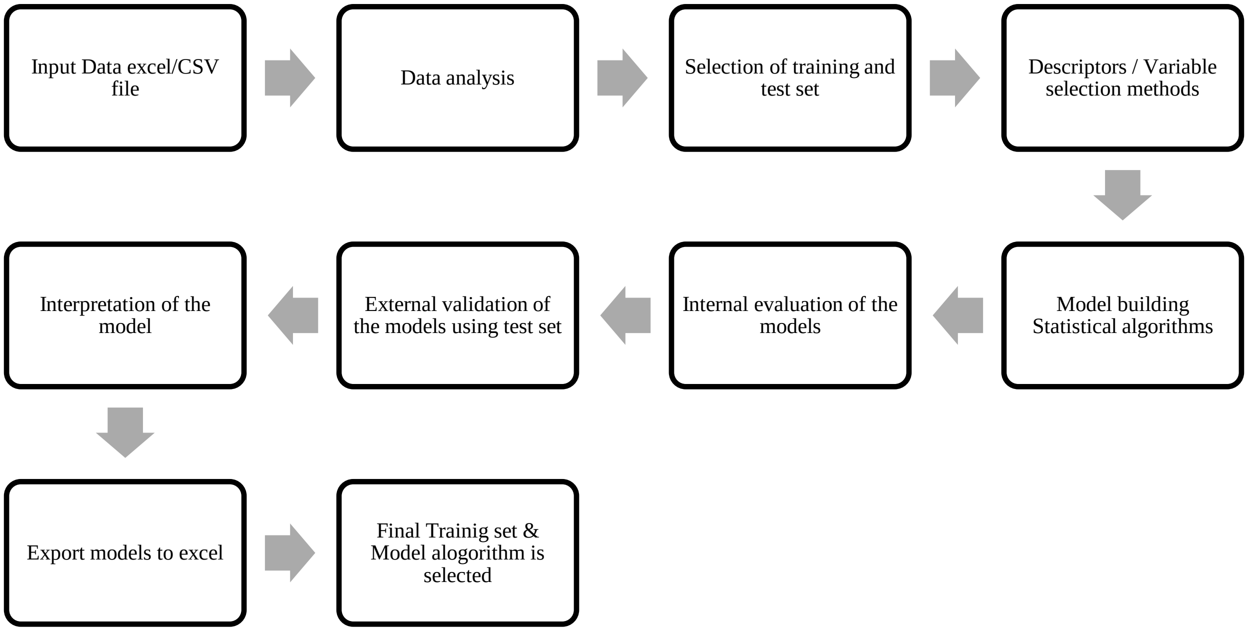

Input Data excel/CSV file
Data analysis
Selection of training and test set
Descriptors / Variable selection methods
Model building Statistical algorithms
Internal evaluation of the models
External validation of the models using test set
Interpretation of the model
Export models to excel
Final Trainig set & Model alogorithm is selected

Fig. 2. Flow Chart Implemented in Automated Machine Learning Program Architecture.

## III. Validation of the Program

The functionality of the program and connection between different flows was verified with two data sets, one for blood-brain barrier (BBB) penetration and the other for Caco2 permeability [20, 21]. Both of these are of two different data types. Viz. are classification based data and quantitative data, respectively. Both of these datasets have published statistical models using manual model building approach. It will be good validation to test the program for reproducibility of published models. The program can also be tested for how automation of model building could benefit the model building for these datasets. It will also help to check if a new algorithm could result in a better model than published.

### *A. Description of the First Dataset*

Dataset1 consists of 1839 molecule entries with fingerprint-based calculated descriptors. Data set is with known blood-brain barrier (BBB) penetration data is collected. The entire dataset was collected from Shen's work[20], which included 1438 BBB+ and 401 BBB- compounds. In the published work by Shen's group has use support vector machine (SVM) algorithm is applied to predict the blood-brain barrier (BBB) penetration. The built model could able to explain the training set and test set with a Q value of 0.9429. Overall, predictive accuracies of the best BBB model for the training and test sets were 98.8 and 98.4%.

### B. Description of the Second Dataset

Dataset2 consists of Caco2 permeability, which is an important parameter, needs to be assessed for estimating the new chemical entities druggability.

Hai Pham The, et al. [21] were managed to general build a QSAR model for caco2 permeability. 21 QSAR models were with discriminate compounds with high Caco-2 permeability (Papp≥8*10−6 cm/s) from those with moderate-poor permeability (Papp<8*10−6 cm/s) were developed on a novel large dataset of 674 compounds. A general model combining all types of molecular descriptors was developed and it classified correctly 81.56 % and 83.94 % for training and test sets, respectively [21].

### C. Model Building for Dataset 1

A dataset with 1839 molecules used to make an automated model building using AML tool (Automated Machine Learning). Total 287 descriptors are calculated using the Chemistry Development Kit (CDK) software.

Data statistics module used to understand the data distribution and the head part of the data. This module is also helpful to understand the data summary like mean, the median of the overall data divided into four quarters. In Fig. 3, output distribution map is depicted which is helpful to understand the dependent variable distribution. In this case, it is categorical distribution i.e. BBB+ or BBB- . For the model, building this kind of data is considered as one and zero internally.

Fig. 4 show cases the primary view of the tool. This shows Data statistics module and results in the side plane. For model building instance 1, data divided into 70:30 percentage for training and test sets respectively. All the classification models implemented in AML tool were applied. In this case, the feature selection module not applied, as this is a classification model building.

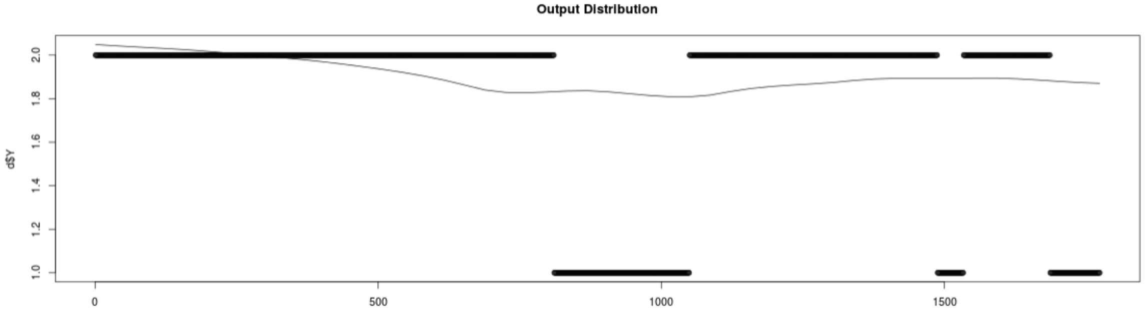
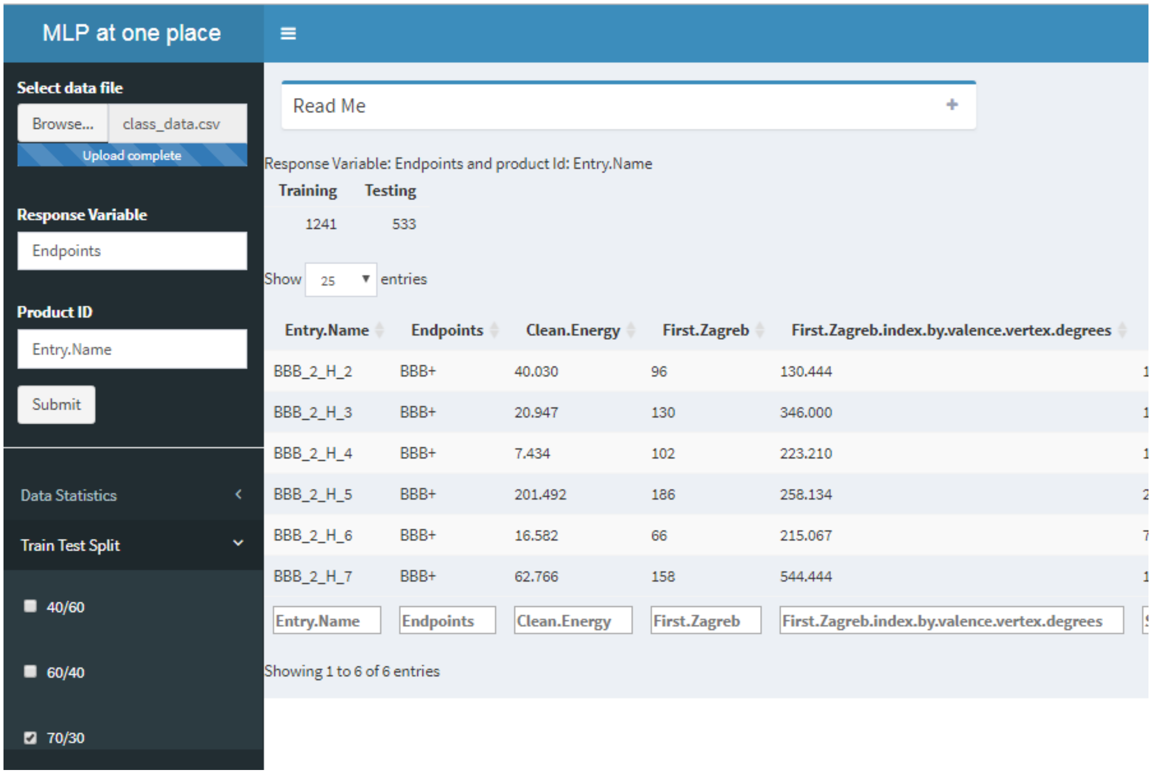


Fig. 3. Output Distribution Map Illustrates Two-Point Distribution of Data.



Select data file

class_data.csv

Upload complete

Response Variable: Endpoints

Product ID: Entry.Name

□ 40/60

□ 60/40

☑ 70/30

Response Variable: Endpoints and product Id: Entry.Name

| Training | Testing |
|---|---|
| 1241 | 533 |

| Entry.Name | Endpoints | Clean.Energy | First.Zagreb | First.Zagreb.index.by.valence.vertex.degrees |
|---|---|---|---|---|
| BBB_2_H_2 | BBB+ | 40.030 | 96 | 130.444 |
| BBB_2_H_3 | BBB+ | 20.947 | 130 | 346.000 |
| BBB_2_H_4 | BBB+ | 7.434 | 102 | 223.210 |
| BBB_2_H_5 | BBB+ | 201.492 | 186 | 258.134 |
| BBB_2_H_6 | BBB+ | 16.582 | 66 | 215.067 |
| BBB_2_H_7 | BBB+ | 62.766 | 158 | 544.444 |
| Entry.Name | Endpoints | Clean.Energy | First.Zagreb | First.Zagreb.index.by.valence.vertex.degrees |

Showing 1 to 6 of 6 entries

Fig. 4. Snapshot of BBB Model Data Selection Page.

Fig. 5 illustrates the multiple models built in less than a min. Model performance with confusion-matrix validation also generated at the same time, including test set performance assessment for different models. Classification model assessment was done using most commonly used methods like confusion matrix, Precision value, and Cohen's kappa coefficient (κ) value.

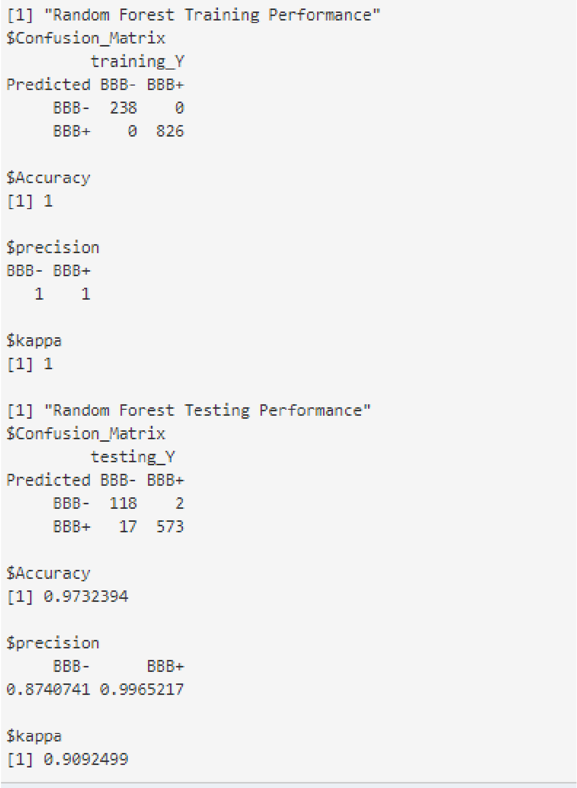
AML tool could able to generate 7 models in a flash of time. The user can pick the best performing model after observing both training and test performance. In this case, Random Forest classification algorithm is best performing with an accuracy of 1 and 0.96 for training and test sets respectively.

In instance 2, data divided into 60:40 percentage training and test sets respectively. Total 1064 molecules are in the training set. All 7 models were generated using AML tool. Again Random forest algorithm could able to explain the data well over SVM methods, KNN and Naive Bayes methods.

Interestingly random forest method accuracy performance was improved to test set 0.97 compared to 0.96 with 30% data in the test set. During instance 3, where training set reduced to 40% has also reduced algorithm test set performance to 0.95 accuracies. Hence, it is identified as Random forest method with 60:40 split of data is giving the good model. In AML tool all these three instances are created, users can able to identify the best algorithm suites to the problem and data set of interest by covering all machine learning space in a few minutes of times.

Total 42 models were generated using AML tool in three instances with a variation in %training set. Random forest algorithm couple with %60 in training set could able to generate the model with %accuracy >80 for both training and tests. Automatic validation and results generation will projected for test set, all models % accuracy is plotted in Fig. 5.

Comparison with published model: When compared the model performance with published models. AML could able to reproduces the equivalent model to the published one, where the test set Q value of 0.9429. The new model developed has test set accuracy of 0.97. Hence, this result validates the performance of the tool and connectivity between workflows and an alternative algorithm to explain the BBB permeation data is established. The earlier publishers did not try random forest models. Fig. 6 shows the Random forest algorithm results in AML tool. Tables I-III compares the different models using different training and test set compositions generated using AML tool.



```
[1] "Random Forest Training Performance"
$Confusion_Matrix
         training_Y
Predicted BBB- BBB+
     BBB-  238    0
     BBB+    0  826

$Accuracy
[1] 1

$precision
BBB- BBB+
   1    1

$kappa
[1] 1

[1] "Random Forest Testing Performance"
$Confusion_Matrix
         testing_Y
Predicted BBB- BBB+
     BBB-  118    2
     BBB+   17  573

$Accuracy
[1] 0.9732394

$precision
     BBB-      BBB+
0.8740741 0.9965217

$kappa
[1] 0.9092499
```

Fig. 5. Model Building Results Output Page with Confusion-Matrix Validation.

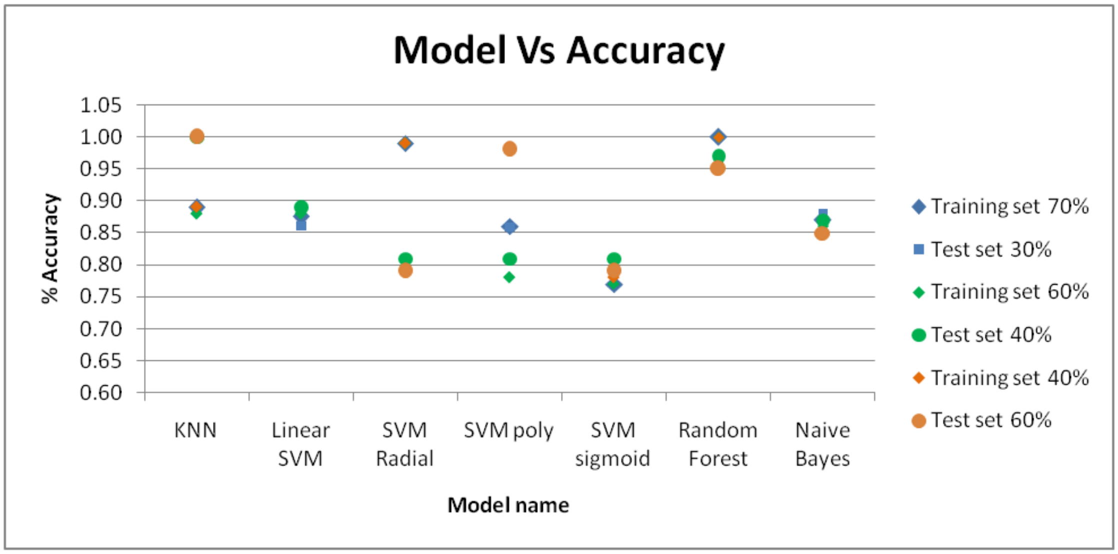


Fig. 6. Multiple Algorithms Predicted Accuracy was Compared Against Training Set Selection. Random Forest Method with 60% Training Set could able to Give Better Accuracy.

TABLE. I. TRAINING AND TEST SET 70:30

| | Training set | | | Test set | | |
|---|---|---|---|---|---|---|
| **Model** | Training set 70% **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** | Test set 30% **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** |
| KNN | 0.89 | 0.729 , 0.946 | 0.70 | 1.00 | 1, 1 | 1.00 |
| Linear SVM | 0.88 | 0.854, 0.882 | 0.67 | 0.86 | 0.765,0.882 | 0.58 |
| SVM Radial | 0.99 | 0.996,1.0 | 0.99 | 0.81 | 0,1 | 0.00 |
| SVM poly | 0.86 | 0.587, 0.946 | 0.58 | 0.86 | 0.489,0.942 | 0.47 |
| SVM sigmoid | 0.77 | 0,1 | 0.00 | 0.81 | 0,1 | 0.00 |
| Random Forest | 1.00 | 1,1 | 1.00 | 0.96 | 0.785,1.000 | 0.85 |
| Naive Bayes | 0.87 | 0.697,0.921 | 0.62 | 0.88 | 0.612,0.949 | 0.59 |

TABLE. II. TRAIN TEST 60:40

| | Training set | | | Test set | | |
|---|---|---|---|---|---|---|
| **Model** | **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** | **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** |
| KNN Training Performance | 0.88 | 0.705, 0.940 | 0.66 | 1 | 1,1 | 1 |
| Linear SVM | 0.88 | 0.689,0.940 | 0.65 | 0.89 | 0.637, 0.949 | 0.62 |
| SVM Radial | 0.99 | 0.995,1.000 | 0.99 | 0.81 | 0.007, 1.000 | 0.01 |
| SVM poly | 0.78 | 0.033, 1.000 | 0.05 | 0.81 | 0.022, 1.000 | 0.03 |
| SVM sigmoid | 0.77 | 0,1 | 0 | 0.81 | 0,1 | 0 |
| Random Forest | 1 | 1,1 | 1 | 0.97 | 0.874, 0.996 | 0.91 |
| Naive Bayes | 0.86 | 0.705, 0.906 | 0.61 | 0.87 | 0.651, 0.930 | 0.59 |

TABLE. III. TRAIN TEST 40:60

| | Training set | | | Test set | | |
|---|---|---|---|---|---|---|
| **Model** | Training set 40% **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** | Test set 60% **Accuracy** | **Precision (BBB-, BBB+)** | **Cohen's kappa coefficient (κ)** |
| KNN | 0.89 | 0.686, 0.948 | 0.66 | 1.00 | 1,1 | 1.00 |
| Linear SVM | 0.57 | 0.773, 0.522 | 0.18 | 0.55 | 0.743, 0.505 | 0.15 |
| SVM Radial | 0.99 | 0.993, 1.000 | 0.99 | 0.79 | 0,1 | 0.00 |
| SVM poly | 0.98 | 1.000, 0.978 | 0.95 | 0.98 | 1.000, 0.985 | 0.96 |
| SVM sigmoid | 0.78 | 0,1 | 0.00 | 0.79 | 0,1 | 0.00 |
| Random Forest | 1.00 | 1,1 | 1.00 | 0.95 | 0.801 0.995 | 0.85 |
| Naive Bayes | 0.85 | 0.753, 0.881 | 0.59 | 0.85 | 0.666, 0.902 | 0.56 |

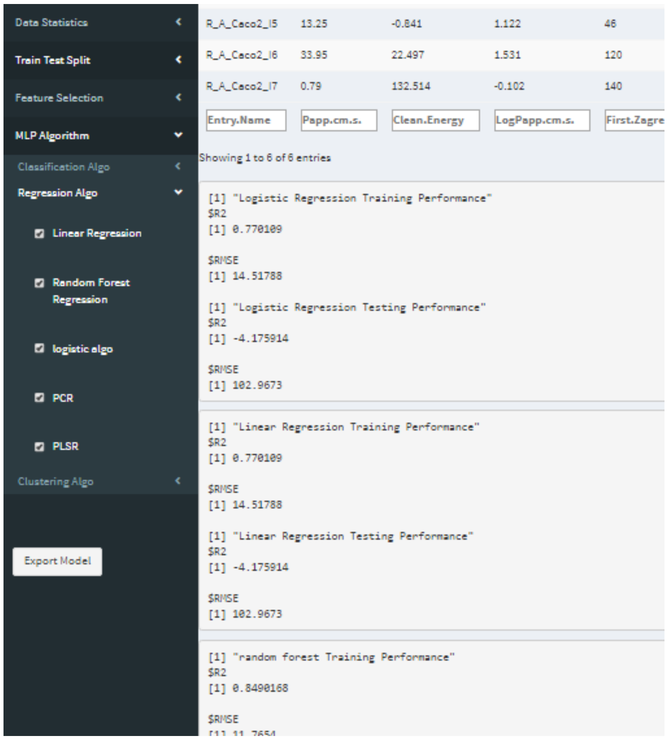
### *D. Model Building for Dataset 2*

Total 674 molecules used to build a regression model for estimating the Caco2 permeability of chemical entities. Descriptors are calculated using CDK software. Using data distribution in AML tool, it was decided to use initially split the data to 60:40 training set and test set.

Total five different algorithms are applied at the same time for model building viz. Linear Regression, Logistic Regression, Random forest, PCR and PLS regression methods which is clearly shown in Fig. 7.

- Caco2 model results.

Random forest algorithm could able to explain training set with R2value of 0.83, with a good RMSE value for test set prediction. Random forest model could able to explain the data much better than the published caco2 model for the same reference set viz. R2 of 0.564, RMSE 0.339. However new descriptors are added to the data set which could improve the model dataset. Tables IV-VI, Fig. 8 and Fig. 9 shows the comparison of other models generated and test set performance results.



| | | | | |
|---|---|---|---|---|
| R_A_Caco2_I5 | 13.25 | -0.841 | 1.122 | 46 |
| R_A_Caco2_I6 | 33.95 | 22.497 | 1.531 | 120 |
| R_A_Caco2_I7 | 0.79 | 132.514 | -0.102 | 140 |
| Entry.Name | Papp.cm.s. | Clean.Energy | LogPapp.cm.s. | First.Zagre |

Showing 1 to 6 of 6 entries

```
[1] "Logistic Regression Training Performance"
$R2
[1] 0.770109

$RMSE
[1] 14.51788

[1] "Logistic Regression Testing Performance"
$R2
[1] -4.175914

$RMSE
[1] 102.9673
```

```
[1] "Linear Regression Training Performance"
$R2
[1] 0.770109

$RMSE
[1] 14.51788

[1] "Linear Regression Testing Performance"
$R2
[1] -4.175914

$RMSE
[1] 102.9673
```

```
[1] "random forest Training Performance"
$R2
[1] 0.8490168

$RMSE
[1] 11.7654
```

Fig. 7. Snapshot of the Tool Shows different Regression Algorithms Generated in AML. Picture also shows Clustering and Classification Algorithm Tabs.

TABLE. IV. TRAIN TEST CACO2_70:30

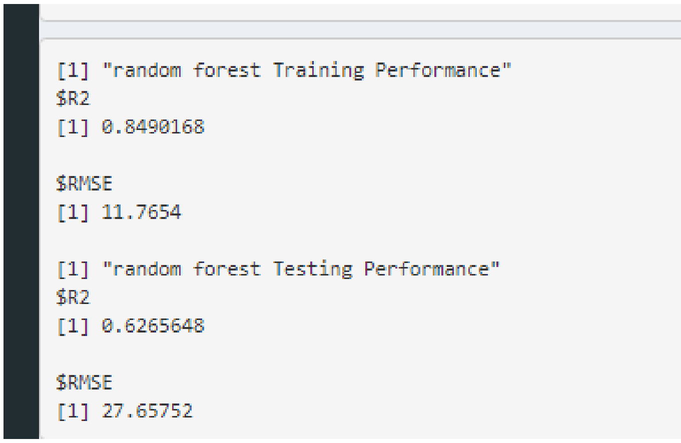
| | Training set 70% | | Test set 30% | |
|---|---|---|---|---|
| **Model** | **R2** | **RMSE** | **R2** | **RMSE** |
| Logistic Regression | 0.77 | 14.52 | -4.18 | 102.97 |
| Linear Regression | 0.77 | 14.52 | -4.18 | 102.97 |
| random forest | 0.85 | 11.77 | 0.63 | 27.66 |
| PCR | 0.32 | 24.88 | 0.32 | 24.88 |
| PLSR | 0.36 | 24.26 | 0.17 | 24.26 |

TABLE. V. TRAIN TEST CACO2_60:40

| | Training set 60% | | Test set 40% | |
|---|---|---|---|---|
| **Model** | **R2** | **RMSE** | **R2** | **RMSE** |
| Logistic Regression | 0.77 | 14.48 | -2.90 | 84.95 |
| Linear Regression | 0.77 | 14.48 | -2.90 | 84.95 |
| random forest | 0.84 | 12.30 | 0.66 | 25.34 |
| PCR | 0.33 | 24.91 | 0.33 | 24.91 |
| PLSR | 0.37 | 24.14 | 0.12 | 24.14 |

TABLE. VI. TRAIN TEST CACO2_40:60

| | Training set 40% | | Test set 60% | |
|---|---|---|---|---|
| **Model** | **R2** | **RMSE** | **R2** | **RMSE** |
| Logistic Regression | 0.80 | 13.75 | -4.38 | 89.70 |
| Linear Regression | 0.80 | 13.75 | -4.38 | 89.70 |
| random forest | 0.84 | 12.13 | 0.69 | 21.25 |
| PCR | 0.32 | 25.21 | 0.32 | 25.21 |
| PLSR | 0.37 | 24.28 | 0.19 | 24.28 |



```
[1] "random forest Training Performance"
$R2
[1] 0.8490168

$RMSE
[1] 11.7654

[1] "random forest Testing Performance"
$R2
[1] 0.6265648

$RMSE
[1] 27.65752
```

Fig. 8. Model Building Results Output Page with Confusion Matrix Validation.

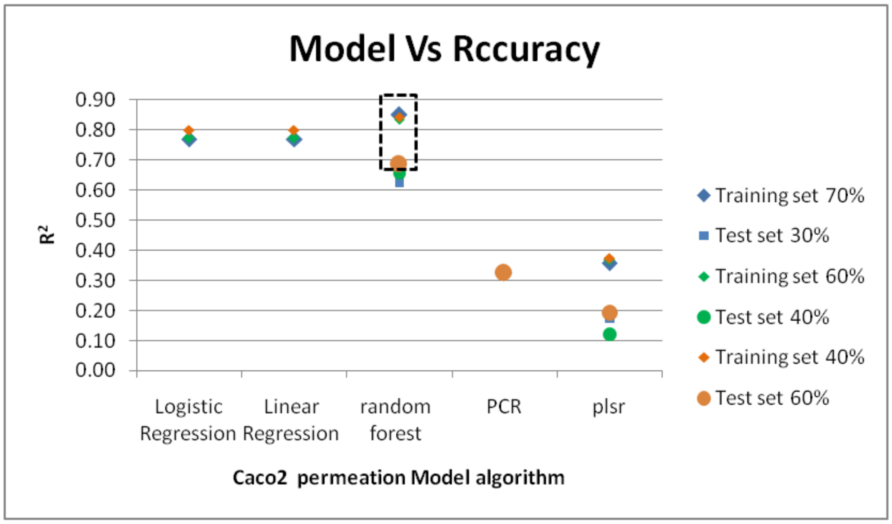


Fig. 9. Multiple Algorithms Predicted Accuracy was Compared Against Training Set Selection. Random Forest Method with 60% Training Set Could Able to Give Better Accuracy.

## IV. Conclusion

Considering multiple machine learning applications, it is recommended to automate the model building process. Automated machine learning (AML) tool is developed and deployed in web portal. The tool is validated for technical capabilities, program stability and tested for seamless connectivity for automating the model building process. The pipeline and interface provides the means to (i) Perform initial data analysis, ii) Identify the data split, (iii) Selection of suitable variable selection method, (iv) Selection of suitable statistical algorithm for model building, (v) Model selection & interpretation. Program is validated against published models and datasets for do-ability and model reproducibility of published models. The tool not only able to reproduce the published results also suggested alternative algorithms, which can explain data variability up to 90% accuracy for training and test sets. Validation carried out on both regression and classification models. AML tool is also tested for potential bugs and abnormal shutdowns. AML tool has potential to generate all kinds of machine learning models at one place; this can be a first place to start with and get an initial combination about data and suitable algorithm to explain the data variations. The workflow is highly flexible, permitting modifications such as a choice of data set, level of theory, validation, or model selection. This can be used for large data sets, by doing the sampling of data from big databases. The tool is hosted in web portal "https://automatedmachinelearningitamcse.shinyapps.io/MLPv3/". The tool can be accessed by anyone who has access to the website.

## References

[1] Ravi Shekar Ananthula, Kishore Madala. (2008) Strategies for generating less toxic P-selection inhibitors: Pharmacophore modeling, virtual screening and counter pharmacophore screening to remove toxic hits. Journal of molecular graphics & modelling. 27(4)546.

[2] Abdelrahman, M. A., Salama, I., Gomaa, M. S., Elaasser, M. M., Abdel-Aziz, M. M. and Soliman, D. H. (2017) Design, synthesis and 2D QSAR study of novel pyridine and quinolone hydrazone derivatives as potential antimicrobial and antitubercular agents. Eur J Med Chem. 138 698-714.

[3] Alpaydin, E. (2010) Introduction to machine learning. MIT Press2nd ed.

[4] Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N. and Lloyd, S. (2017) Quantum machine learning. Nature. 549(7671)195-202.

[5] Galvan, R. A. R. M. J. M. V. I. M. A Study of Machine Learning Techniques for Daily Solar Energy Forecasting Using Numerical Weather Models. Intelligent Distributed Computing VIII. DOI: 10.1007/978-3-319-10422-5_29269-278.

[6] Gong, G. I. J. L. M. C. H. H. Z. Z. P. (2019) A review on machine learning methods for in silico toxicity prediction. Journal of Environmental Science and Health, Part C 36(4)169-191.

[7] RT., M. G. S. (2019) A user-generated data based approach to enhancing location prediction of financial services in sub-Saharan Africa. Appl Geogr.105 25-36.

[8] CH, L. S. L. J. C. Y. Y. H. K. J. J. (2019) Machine learning models based on the dimensionality reduction of standard automated perimetry data for glaucoma diagnosis. Artif Intell Med.94 110-116.

[9] K, Z. M. T. Y. K. H. H. (2018) Machine Learning With K-Means Dimensional Reduction for Predicting Survival Outcomes in Patients With Breast Cancer. Cancer Inform.17 1-7.

[10] modgen. https://www.statcan.gc.ca/eng/microsimulation/modgen/modgen.

[11] Eggensperger, M. F. A. K. K. (2015) Efficient and Robust Automated Machine Learning. Advances in Neural Information Processing Systems 28 (NIPS 2015)1-9.

[12] Jordan, M. I. (2015) Machine learning: Trends, perspectives, and prospects. Science255-260.

[13] Koohy, H. (2018) The rise and fall of machine learning methods in biomedical research. F1000Research. 6 1-16.

[14] Vink, R. M. S. P. L. G. (2018) Generating missing values for simulation purposes: a multivariate amputation procedure. Journal of Statistical Computation and Simulation. 88(15)2909-2930.

[15] De Silva AP;, M.-B. M. D. L. A. L. K. S. J. Multiple imputation methods for handling missing values in a longitudinal categorical variable with restrictions on transitions over time: a simulation study. BMC Med Res Methodol. 19(1).

[16] Kang, H. (2013) The prevention and handling of the missing data. Korean J Anesthesiol.64(5)402-406.

[17] Hogan, C. J. H. L. E. C. J. W. (2015) Are all biases missing data problems? Curr Epidemiol Rep. 2015 Sep 1; 2(3): 162–171.2(3)162-171.

[18] JJ, K. H. S. S. G. R. S. S. G. (2016) Dimension reduction and shrinkage methods for high dimensional disease risk scores in historical data. Emerg Themes Epidemiol. 2016 Apr 5;13:5. doi: 10.1186/s12982-016-0047-x. 13 5.

[19] Hawkins, D. M. (2004) The Problem of Overfitting. J. Chem. Inf. Comput. Sci.44(1)1-12.

[20] Shen, J. C., F.; Xu, Y.; Li, W.; Tang, Y.;. (2010) Estimation of ADME properties with substructure pattern recognition. J Chem Inf Model. 50(6)1034-41.

[21] Pham The, H., Gonzalez-Alvarez, I., Bermejo, M., Mangas Sanjuan, V., Centelles, I., Garrigues, T. M. and Cabrera-Perez, M. A. (2011) In Silico Prediction of Caco-2 Cell Permeability by a Classification QSAR Approach. Mol Inform. 30(4)376-85.