WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Stacking Regression-Based Model for Predicting Patient's Length of Stay in a Semi Urban Hospital

Héritier Nsenge Mpia[1], Moïse Katembo Kasolene[2], Vingi Mutegheki Baraka[3], Nephtali Inipaivudu Baelani[4]

[1]Department of Informatique de Gestion, Université de l'Assomption au Congo, Butembo, DR Congo
[2]Faculty of Applied Sciences, Université de l'Assomption au Congo, Butembo, DR Congo
[3]Department of Informatique de Gestion, Université de l'Assomption au Congo, Butembo, DR Congo
[4]Department of Informatique de Gestion, Université de l'Assomption au Congo, Butembo, DR Congo

DOI: https://ijrpr.com/uploads/V4ISSUE2/IJRPR9896.pdf

Abstract

In this research, the authors found that statistical analysis is very important preliminary phase in Machine Learning, especially for regression problems. Indeed, when the authors developed the first single models using the same algorithms and the same dataset, they obtained poor performances. After verifying the assumptions of the multiple linear regression, they adjusted the used data and produced efficient models. Moreover, as the objective was to apply the stacking model to predict Patient's Length of Stay in a semi urban hospital, the results showed that the stacking regressor performed better than the seven different models implemented (Random Forest, Extra Trees, Decision Tree, XGBoost, Multilayer perceptron, Light GBM, Support Vector Regressor (SVR)) taken individually. The authors combined Random Forest Regressor, Extra Trees Regressor, Decision Tree Regressor, XGBoost, Light GBM, and SVR to build the stacking model. Using secondary data from four services (Pediatrics, Hospitalization, Gynecology, and Neonatology) of a semi-urban hospital, located in a region of ongoing war in eastern Democratic Republic of Congo (DRC), the study examined the minimum length of stay of a patient in hospital when admitted in one of the four above services. Performances were evaluated using MAE, RMSE, MSE, R-squared and Accuracy. The stacking regression model shifted from 85% of accuracy before statistical analysis phase to 91% after applying statistics and from 0.75 to 0.91 as R-squared.

Keywords: *Length of stay in hospital, Ensemble method, Optimization, Statistical analysis, Stacking regression*

1. Introduction

The length of stay (LOS) of patients is a very important component that helps the hospitals plan operations. The lower the stay in hospital the higher the bed turnover rates which leads to increase [1]. In this paper, we used height different Machine Learning (ML) algorithms in order to evaluate which model predicts better the LOS of Patients in a semi urban hospital. Seven of them were combined using stacked ensemble regression from mlxtend library and Deep Neural Network regression was the seventh one.

The following objectives constituted the core of this research:

i. To perform a brief empirical review of the recently published papers on LOS of Patients, which applied techniques for regression ML tasks

ii. To optimize LOS prediction by preparing statistically our collected data before starting ML phase

iii. To build a stacked ensemble regression by combining XGBoost Regressor, Light GBM, SVR, Decision Tree Regressor, Extra Trees Regressor, and Random Forest Regressor techniques.

iv. To compare Deep Neural Network regression model and the stacked regression model

v. To examine and compare accuracy, MSE, MAE, RMSE, R2 metrics of the used techniques over Patient's discharge data from a semi urban Hospital in eastern of DRC.

Three questions constituted the cognitive effort of this study. Those questions are:

(i). How can statistical analysis help to improve ML model performance?

(ii). Are the used features to predict LOS statistically significant?

(iii). Which ML regression model predicts better LOS in a semi urban hospital?

2. RELATED WORKS

Oliveira, et al. [2] developed a ML model to predict the number of patient discharges per week using data mining techniques. They used classification techniques using Naïve Bayes, SVM and Decision Trees as algorithms. The services involved were orthopedics, childbirth, nursery and obstetrics. They obtained an accuracy ranging from 82.69% to 94.23%. Kavanaugh, et al. [3] conducted research to predict length of stay in a child psychiatric hospitalization program. They had a sample of 96 children with ages ranging from six to fourteen years. They analyzed the influence of neuro cognition on subsequent LOS using correlation and linear regression as models. Using the ROC curve, these researchers were able to detect that the global deficit score used as a performance measure was able to distinguish children who may or may not have a longer hospital stay.

Grampurohit and Sunkad [4] used Ridge, Lasso, Linear regression and ElasticNet to predict the LOS of a patient from the day of his/her admission to the hospital to the day of his/her discharge. These researchers used Mean Absolute Error (MAE) to evaluate their models. Their results revealed that the linear regression performed poorly due to the overfitting presence in the model with a MAE of 198379877732011.9. While Lasso gave a MAE of 0.96865, ElasticNet had a performance of 0.95121 and Ridge had the best score with an MAE of 0.82131.

Bassam, et al. [5] conducted a study on 2017 patients with COVID-19 admitted to Rashid Hospital, Dubai from January 1, 2020 to July 20, 2020. The researchers used decision tree to predict LOS and R-squared, accuracy, sensitivity and specificity as metrics to assess the performance of the model. According to their results, R-squared was 49.8%. The accuracy was 96%. The sensitivity was 96.5 and the specificity 87.8%. According to them, the mean absolute deviation of hospital stay was 3.9 days while the median absolute deviation indicated a value of 2.85 days.

Using datasets containing Covid-19 patient data, Sinha, Tushar, and Goel [6] used catboost as algorithm to predict the maximum duration of a patient's stay in a hospital. Their goal was to help hospitals quickly manage hospital resources such as beds and medications. This allows hospitals to be productive and systematic in their operations while maximizing the time to discover a patient's infection in order to provide rapid treatment and optimize the patient's stay in the hospital, especially in times of Covid-19. Their model gave an accuracy performance of 92.33% for the train set.

3. MATERIALS AND METHODS

The data used for this study was collected at Clinique La Lumière, located in the city of Butembo, North Kivu Province, DR Congo. This is in fact secondary data. Since Clinique La Lumière is located in a semi-urban area, there is still the inadequacy of the use of information systems to store the daily data of patients. Thus, it is from the patient registry that that data was extracted. 838 samples were extracted and these samples vary from the period of January 2, 2010 to August 25, 2021. The dataset was in French but the authors translated it into English and it included seven attributes: Patient ID, Date of Entry, Date of Discharge, Gender, Age, Disease, and Service where the patient was admitted. This clinic has four different departments: Neonatology, Gynecology, Hospitalization and Pediatrics. The distribution of patients in these four different departments is shown below:

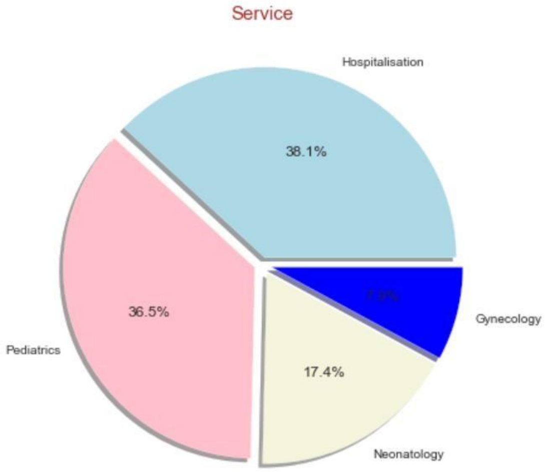


Figure 1. Distribution of Patients by Service

As it can be seen in the above figure, Hospitalization had the highest record followed by Pediatrics and Neonatology. Only by looking at the services of this clinic and the number of patients in each service, it can be concluded that this clinic is mainly dedicated to the integral care of women and children. Thus, the main objective of this study was to predict, using data mining techniques, the number of days that a patient can spend in the different departments of the Clinique la Lumière. The authors wanted to ensure the adequate management of hospital beds, especially since several diseases including the Ebola epidemic threaten the region in which the clinic is located. Hence, the need to carefully manage the entries and discharges of the hospital to avoid the congestion of sick children in this semi-urban clinic.

3.1. Study design

Three architectures were proposed in order to achieve the objectives of this study. First, the authors defined the stacking regression pipeline. Secondly, they built the Deep learning model, as one of the objectives was to compare the performance of the stacked regression model and the Deep Neural Network.

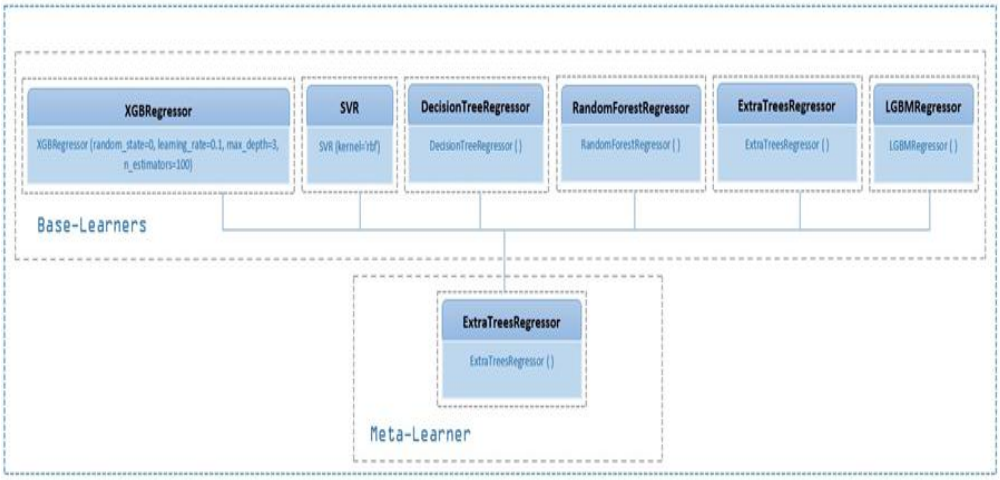


Figure 2. Stacked ensemble regression pipeline

Known as one of the powerful techniques for prediction [7], stacking ensemble helped the authors to build an accurate model for predicting Patient's length of stay. The above pipeline (Figure 2) illustrates the architecture of the developed stacked regression model. The authors did not tune neither the base-learners nor the meta-learner. In fact, when applying GridSearchCV, the authors were getting a lower performance for both single and ensemble models. Therefore, only by tuning extreme gradient boost, with random state equals zero, learning rate of 0.1, three as max_depth, and n_estimators equals 100 and setting up radial basis function as the kernel of SVR to reduce the dimensionality of the data [7], the model came up with a higher performance and reduced RMSE, MAE, and MSE metrics and increased the coefficient of determination.

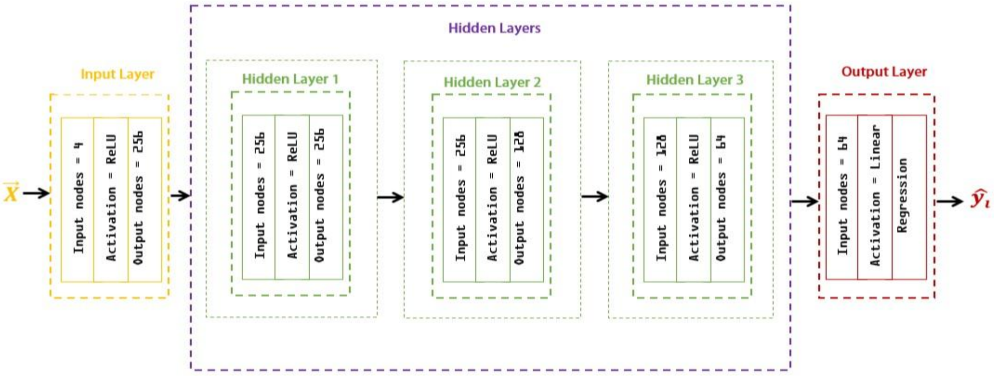


Figure 3. Deep Neural Network Pipeline

The above figure 3 illustrates the components of the developed Deep Neural Network model. The author created this model using Keras Sequential model. The model has the following architecture:

→ The first dense, the input layer has four as input dimensions as we have 4 features and 256 output nodes. To activate the output, we used ReLU.

→ The study proposed three hidden layers both using ReLU as activation function.

→ The output layer was a fully connected layer with 64 input nodes from the output of the last hidden layer and the output was a regressive value. Since the problem is a linear-based regression, the author did not use a nonlinear function for the activation [8], but rather a linear one such that at the output, the value of the neuron is equal to its activation level, in which the function does not change its value [9]. This function can be written as follows:

$$y = \sum_{j=1}^{n} w_j x_j$$

On the other hand, the entire proposed architecture of this study is illustrated below. The objective of this architecture was to predict the LOS of Patients based on four explanatory variables: Gender, Age, Disease, Service. The study was divided into five main steps: Data pre-processing, Statistical Analysis, Data Transformation, ML Tasks, and Evaluation of model performances. Figure 4 presents the illustration of these steps in a schematic way. Since the use of secondary data leads researchers to rely on the original measurement tool used to collect primary data [10], it was difficult for the authors to evaluate the instrumentation used to collect these data. Hence, the intervention of data pre-processing phase in order to not only deal with bias, missing data but also to perform feature engineering.

The statistical phase allowed the authors to extract relevant and reliable information for ML operations. The impact of this phase was beneficial because it made the authors learn about the phenomena observed [11] in the collected data and know if the secondary data obtained are favorable for a linear regression. Consequently, the importance of the transformation phase to overcome the assumption of normality for a good regression analysis and modeling. Before performing ML tasks, the authors split first the cleaned dataset by using 101 as random state value, 25% for test set and 75% for train set. At the end, the study evaluated the metrics of the different built models.

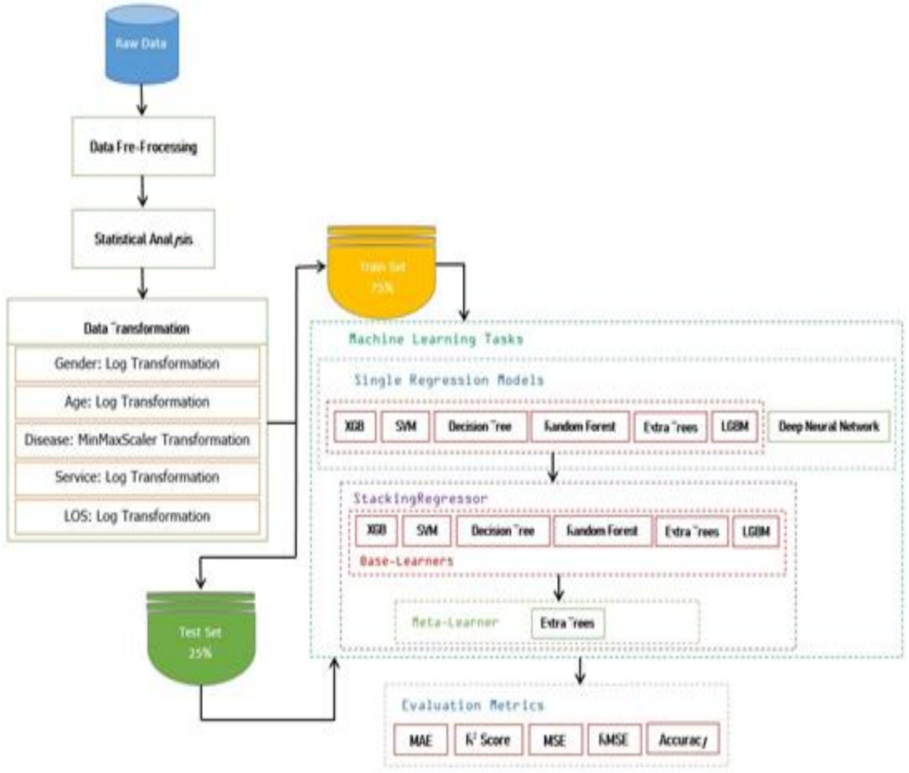


Figure 4. Study Architecture

3.2. Data preparation

As the dataset came from a register, it had several internal issues. The biggest issue was that some dates of discharge were put in the column of entry date. Therefore, when trying to get the LOS values by computing date of discharge minus date of entry, results were negative values. To handle this issue, after getting the new column of LOS, by performing feature engineering, the authors used the function *abs()* from Pandas in order to convert all the obtained LOS to absolute values. The second issue was the column Age. As the Clinic has a Neonatology Department, that column comprised ages of patients measured in Year and others in Days. To uniform data and use only Year as a unit, the authors split first the dataset into two parts. In the first part, he put all data where Age does not have Day. In the second part, it was put only data where the column Age contains Day as a part of the age. After that, values were extracted in the column Age of the second part using *str.extract()* from Pandas [8] in order to get only the numeric part of the date and then convert it into Year by dividing the age by 365. Finally, the authors combined the two dataset. They obtained thus a dataset that has only Age in Year.

To make the used data a regression problem, the authors dropped Patient ID, Date of entry, and Date of discharge as the feature-engineering step already allowed obtaining LOS as a dependent variable from Date of entry and Date of discharge. Moreover, the authors used *LabelEncoder()* from *sklearn.preprocessing* in order to encode all the data of the column Disease in numeric and *map()* function from Pandas to replace string values in the Service column by the numeric values.

After performing the statistical analysis by transforming the data and removing outliers, the final dataset contained 706 records. At the end, the used dataset to perform ML tasks contains transformed data. Hence, to use this data for another purpose, one would inverse the transformation. For Gender, Age, Service and LOS, to reverse the transformation, one needs to use *np.expm1()* function while for Disease, one needs to apply *scaler.inverse_transform* function. However, the following table illustrates the description of variables and presents the code format of the data:

Table 1. Final data structure before log and MinMaxScaler transformations of variables

No	**Variable**	**Code format**
1	Gender	0: F, 1: M
2	Age	All ages were encoded using the pythonic *log1p()* function. The range of age is from one day to 65 years old. The unit is year. Therefore, babe ages (days) were converted by year (days/365).
3	Disease	0: Abortion, 1: Acute Gastritis, 2: Acute bronchitis, 3: Acute encephalitis, 4:Acute enteritis, 5:Acute febrile gastroenteritis, 6:Acute gastroenteritis, 7:Afebrile, 8:Agnosia, 9:Allergy, 10:Appendicular, 11:Arteriovenous malformation, 12:Asphyxiated preemie, 13:Bilateral hernia, 14:Bronchitis, 15:Chronic valvular disease, 16:Cold and bronchitis, 17:Diabetes, 18:Digestive tract, 19:Dyspeptic and emotional shock, 20:Emotional trauma, 21:Extreme prematurity, 22:Facial fall, 23:Febrile enteritis, 24:Febrile gastroenteritis, 25:Fever, 26:Flu, 27:Flu and malaria, 28:Focal Cortical Dysplasia, 29:Gastritis, 30:Hernia, 31:Hypertension crisis, 32:Hypothermia, 33:Hypotrophy, 34:Incomplete abortion, 35:Infection, 36:Infection and Diabetes, 37:Infectious Mononucleosis, 38:Infectious risk, 39:Influenza, 40:Losartan effect, 41:Low birth weight, 42:Macrosomia, 43:Malaria, 44:Medium prematurity, 45:Mocking seizure, 46:Neonatal asphyxia, 47:Neonatal infection, 48:Obstetrical pathology, 49:Pelvic trauma, 50:Peptic ulcer, 51:Polytrauma, 52:Post-partum infection, 53:Pregnancy discomfort, 54:Probable acute cystitis, 55:Probable late vision, 56:Respiratory distress, 57:Rheumatism, 58:Right Ovarian Cyst, 59:Salmonellosis, 60:Senile Dementia, 61:Sepsis, 62:Severe malaria, 63:Short-term fever, 64:Threat of preterm birth, 65:Uterine infection, 66:Uterine myoma, 67:Vaginal mycoses
4	Service	0: Pediatrics 1: Hospitalization 2: Gynecology 3: Neonatology
5	LOS	This is the target. Values of this continuous variable are from the subtraction of Discharge date and Entry date

3.3. Predictive used models

XGBoost: XGBoost, which means eXtreme Gradient Boosting, is open-source ensemble technique applicable in regression and classification and seems to be a kind of instance of the Gradient Boosting Machine algorithm [12]. As an ensemble technique, XGBoost allows the creation of new models to correct the residuals of the used models in the ensemble, thus combining their results to obtain the final optimal prediction [13]. It is a faster algorithm than other assemble estimators. This has made it the ideal and popular algorithm in data mining and ML [14].

Support Vector Machine: SVM belongs to the family of supervised algorithms and is applicable in both regression and classification problems. The algorithm is built using some parameters with a kernel function that can be linear, Gaussian or polynomial [15]. SVM is one of the powerful algorithms for solving nonlinear problems and is effective when a small sample size is available. However, the performance of the SVM model often depends on the kernel used and the appropriate selection of the penalty parameters [16].

Decision Tree: The decision tree is an algorithm that operates as a tree such that any path from the root of the tree is described sequentially by a separation of the data until a boolean result is obtained in the leaf node [17]. It is an algorithm that optimizes the model well because it generalizes due to its predictive output that is presented in a hierarchical manner [18]. The decision tree is much more widely used because it gives even non-ML specialists the ability to interpret easily the results from the tree and ensures the stability of outliers found in the dataset [15].

Random Forest: Random Forest is a very flexible ML algorithm, applicable to both regression and classification problems. In its operation RF builds various decision trees during the training to produce an average prediction of all the decision trees that have been generated [19]. This algorithm was originally developed to combine several trees in the two different types of problems mentioned above by referring to CART. This combination is done using the bagging method. Thus, the algorithm is a set of trees each depending on random variables [20]. For the regression cases, RF uses a continuous target variable and, in the classification, there is presence of target variable of categorical type [19].

Extra Trees: Extra-tree is an ensemble algorithm based on a tree. This algorithm extends the random forest algorithm. In its operation, extra-tree regression constructs an ensemble of a regression tree that is not pruned using a classical descending procedure [21]. This recent approach goes beyond the random forest functionalities. It uses a random subset of features to construct each base estimator [22].

Light Gradient Boost Machine: LGBM is an ensemble-learning algorithm. Here, the model is built sequentially by minimizing the error of the previously learned models iteratively [23]. This algorithm has the advantage of being easy to implement, to understand, and to reduce the complexity of the model [24]. According to Hou et al., [25], LGBM is more advantageous than XGBoost and GBDT in terms of regression fit computation.

Deep Neural Network: Deep learning usually uses a multi-layer network using the gradient algorithm to build efficient models [26]. It is a learning algorithm best suited for large masses of unstructured data [27], which contains input layers with some independent variables and parameters, hidden layers capable of transferring the information from the input layer to the output layer [28] using an activation function such as the sigmoid function [29].

Stacking ensemble: Using Stacking as a method provides better performance because it combines the predictive efforts produced by base learners at the first level and then applies a layer called meta-learner to try to combine the previous results of the base learners to ensure the generalization of the model. Stacking aims to stack the predictions of various models by applying the linear combination of weights of base learners [8].

3.4. Used Tools and Software

Anaconda was used as an environment to develop the models used in this study. To perform statistical analysis, the author used norm library from *scipy.stats* and the numpy function log1p which applies log(1+x) [30] was applied to all data of Gender, Age, Service and of the target LOS in order respectively to transform the mentioned variables. *MinMaxScaler()* from *sklearn.preprocessing* was used to transform the Disease variable. To detect outliers in data, authors used boxplot from Pandas library [31]. Numpy helped, through its function percentile, to replace all the outliers by *np.nan()*. Later on, the authors dropped all the NaN rows. To build a linear model, which helped the author sto verify the relationship between the fourth independent variables and the dependent variable of this study, ordinary least squares (OLS) was used as library from *statsmodels.formula.api*. This library was useful because it allowed the authors to verify some regression assumptions such as the significance of features, homoscedasticity, normality, independence, and the multicollinearity [32].

On the other hands, regarding the ML phase, the authors used XGBRegressor from xgboost open source library to perform the first single regression model; SVR from sklearn.svm; DecisionTreeRegressor from sklearn.tree; RandomForestRegressor from sklearn.ensemble; ExtraTreesRegressor from sklearn.ensemble; LGBMRegressor from lightgbm library; and Sequential model from Keras to build the Deep Neural Network regression. At the end, modelStackingRegressor from mlxtend.regressor library was used to perform stacked ensemble regression.

Moreover, residuals_plot from yellowbrick.regressor was used in order to plot the residuals of the stacking regressor model. To plot metrics results of the used models, the authors used Excel 2016.

3.5. Performance metrics

As shown in Figure 3, the study applied five different metrics to evaluate the performance of the models. The first metric was the MAE. This metric is considered in statistics as the average magnitude of the errors obtained in a set of predictions, regardless of their direction [33]. In other words, on a sample tested in a statistical analysis, MAE is the average of the absolute differences between the predicted value and the actual observation, while assigning the same weight to all individual differences [34]. The following formula illustrates how to calculate the MAE:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

where *n* is the number of samples, y_i the actual observation and $\hat{y}_i$ the predicted value.

Moreover, R-Squared was the second metric to be used. The coefficient of determination R2- Squared is the proportion of the variance that can be found in the variable to be explained or dependent variable in regression that is predictable from the explanatory variables [35]. The formula to perform this metric is as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(X_i - Y_i)^2}{\sum_{i=1}^{m}(\bar{Y} - Y_i)^2}$$

Where X_i represents the predicted value, Y_i the actual value and $\bar{Y}$ the mean of all the actual values. The authors also used the Mean Squared Error (MSE) to assess the performance of the models. This metric, in statistics, is defined by:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

MSE is much more useful to compare several models, especially when one of these estimators is biased. In the case where the estimators to be compared are unbiased, the best performing estimator will be the one with the smallest variance. Therefore, the MSE can be expressed as a function of the bias of an estimator and its variance [36].

The fourth metric used in this study was the root mean square error (RMSE), which is defined mathematically via the equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Strictly speaking, the RMSE is a measure of the average error, weighted by the square of the error. The purpose of this metric is to provide knowledge of the magnitude of the error of the prediction by not indicating the direction of the errors. RMSE is a squared quantity that is more influenced by large errors than by small errors. Thus, having a small RMSE value determines the higher accuracy of a model [37].

4. RESULTS AND DISCUSSIONS

RQ1. How can statistical analysis help to improve ML model performance?

Although it is known that ML is totally different from statistics, such as for statistics, there are assumptions to build a model, while ML process does not often have assumptions [38], the authors of this study proved the necessity of a primordial phase of statistical analysis when it comes to secondary data and linear regression before moving on to ML tasks [39]. This statistical phase allowed to settle the adequate data and to get appropriate explanatory variables to be fed in the ML model thanks to the verification of the linear regression assumptions. Therefore, after verifying these assumptions, the author was able to build a powerful stacked ML model that reduces MAE, MSE, R-Squared and RMSE.

Actually, the multi linear regression assumptions can be expressed mathematically as follows:

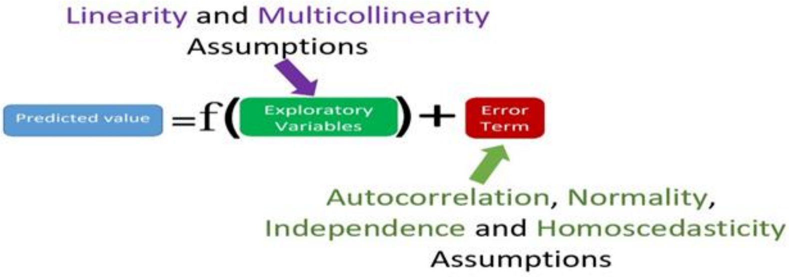

Linearity and Multicollinearity
Assumptions
Predicted value
=f(
Exploratory Variables
)+
Error Term
Autocorrelation, Normality,
Independence and Homoscedasticity
Assumptions

Figure 5. Representation of multi linear regression assumptions

Multicollinearity among features: After performing Variance Inflation Factor (VIF) to check the multicollinearity of the explanatory variables, the authors obtained VIF of 1.89 for Gender, 2.14 for Age, 3.82 for Disease, and 2.29 for Service. There was no feature with VIF more than 5. Hence, there was no significant multicollinearity or a moderate correlation between the regressors [40].

Linearity and Normality Test: At the first look, the used dataset was not normally distributed. Therefore, the authors applied *log1p()* and *MinMaxSclaer()* functions in order to fix the problem of multivariate normality. After applying these two functions in the variables, the authors were able to verify the distribution of the data through the histograms below:

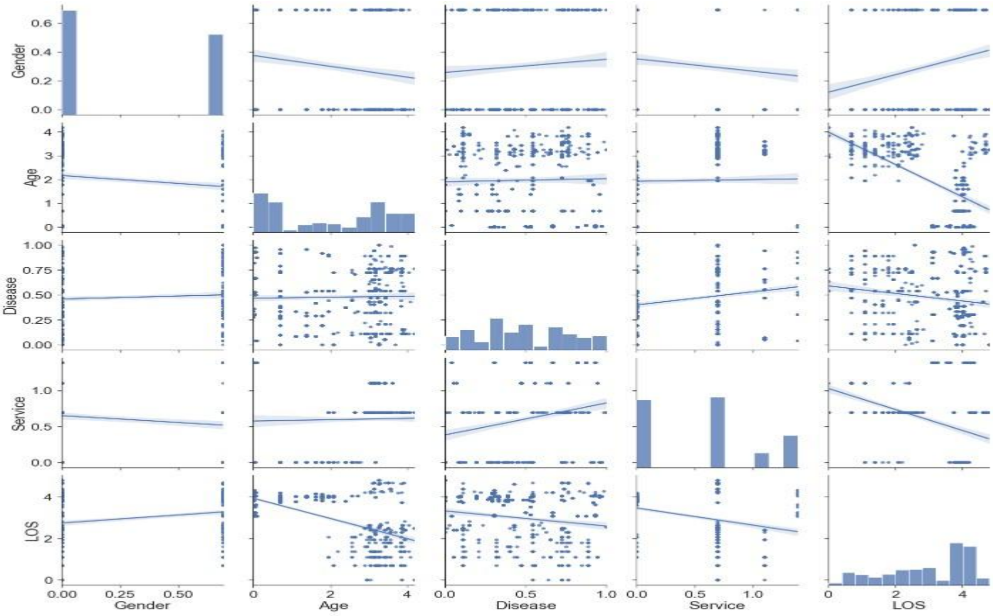


Figure 6. Linearity and Normality after transformation

Autocorrelation of residuals: Since there are some peaks outside the gray zone of the confidence interval, in the below plot, the authors concluded that there is some autocorrelation in some lags between our residuals. This can be seen in table 2 where the Durbin-Watson test is 0.315, which is less than 2. Hence, there was a positive correlation between the residuals [41]. Thus, the violation of autocorrelation of residuals.

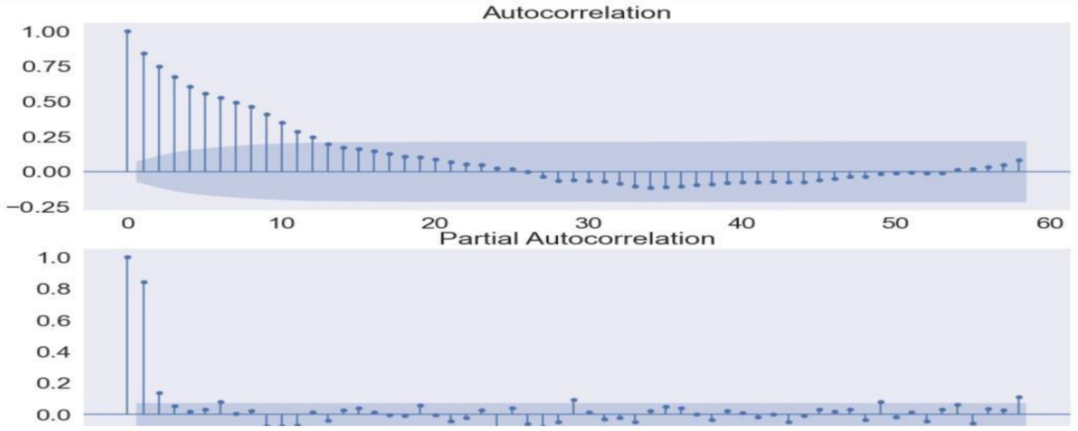


Figure 7. Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) graphs of residuals

In fact, a model with auto-correlated residuals symbolizes the dependence of previous and current observed values. This means that there are unexplained underlying factors in the dependent variable that are in the error terms [42].

Homoscedasticity: From the below scatter plot, it can be seen that when the predicted values are less, the residuals are also less, though it can be seen on the right when the predicted values are big, the residuals values tend to be less. To confirm the homoscedasticity, the authors also computed the correlation of predicted values and residuals and we got -0.0. This confirmed the assumption of homoscedasticity. Because the correlation between the predicted values and the residuals was not significant, the authors concluded that the assumption of homoscedasticity was satisfied.

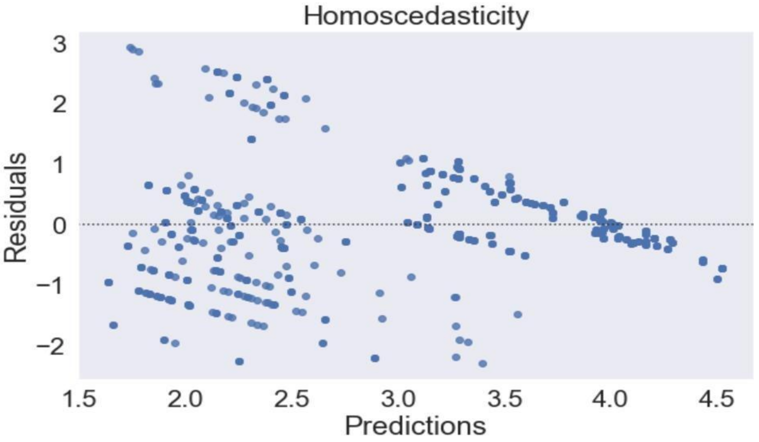


Figure 8. Homoscedasticity

Before performing statistical analysis, the built stacked regression model had an accuracy of 85%, MAE of 15.97, R-Squared of 0.85, MSE of 3163.1 and RMSE of 56.24. However, after checking the assumptions and transforming our data, the stacked model performed better as it can be seen in Table 4.

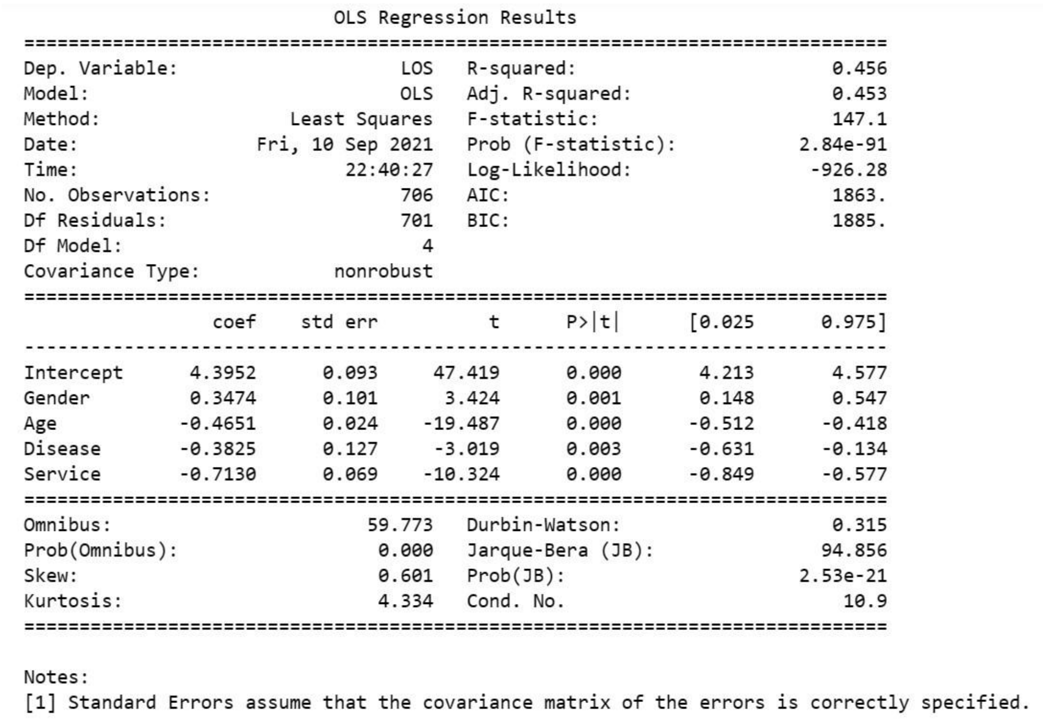
RQ2. Are the used features to predict LOS statistically significant?

The above left part of table 2 returns the dependent variable (LOS), the number of observations (706 rows) in the processed data set and the degree of freedom of the regression model (Df Residuals). Recall the Df Residuals is calculated as follows:

Df Residuals = No. Observations - Df Model - 1

where No. Observations are 706 rows and Df Model number of our predicting variables, that is 4. Moreover, this model has a nonrobust covariance type, which means that the model did not use the robust covariance, which aims at minimizing or even eliminating all positively, or negatively related variables [43].

Table 2. OLS Results for the regression model for verifying the significance of used features



```
                            OLS Regression Results
==============================================================================
Dep. Variable:                    LOS   R-squared:                       0.456
Model:                            OLS   Adj. R-squared:                  0.453
Method:                 Least Squares   F-statistic:                     147.1
Date:                Fri, 10 Sep 2021   Prob (F-statistic):           2.84e-91
Time:                        22:40:27   Log-Likelihood:                -926.28
No. Observations:                 706   AIC:                             1863.
Df Residuals:                     701   BIC:                             1885.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      4.3952      0.093     47.419      0.000       4.213       4.577
Gender         0.3474      0.101      3.424      0.001       0.148       0.547
Age           -0.4651      0.024    -19.487      0.000      -0.512      -0.418
Disease       -0.3825      0.127     -3.019      0.003      -0.631      -0.134
Service       -0.7130      0.069    -10.324      0.000      -0.849      -0.577
==============================================================================
Omnibus:                       59.773   Durbin-Watson:                   0.315
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               94.856
Skew:                           0.601   Prob(JB):                     2.53e-21
Kurtosis:                       4.334   Cond. No.                         10.9
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

P >|t| of Gender is 0.001, that of Age is 0.000, of Disease 0.003, and of Service is 0.000. Both are smaller than 0.05, the p-value. These are good values if one wants to have 95% confidence in the findings of the model. It means that there are respectively 0.1%, 0%, 0.3%, 0% chances that the four predicting variables have no effect on the LOS which is the dependent variable. Clearly speaking, those exploratory variables influence the LOS significantly. In that case, the author rejected the null hypothesis. It means that those four variables have an influence on LOS. The increasing or decreasing of one of them will influence the LOS. The findings presented a value of 0.456 as R-squared. Therefore, the authors concluded that Genre, Age, Disease, and Service explain about 45.6% of the variability in LOS. Performing the estimation procedure as shown above, the author concluded that LOS and Gender, Age, Disease, and Service are related such that:

LOS = 4.3952 + (0.3474 * Gender) + [(-0.4651) * Age] + [(-0.3825) * Disease] + [(-0.7130) * Service]

where 4.3952 represents the Intercept, which gives the response variable when other variables are equal to zero or constants [44]. Table 3 below shows the significance of all the features, which were obtanied after performing the *pvalues()* function from the fitted model as follows fitted_model.pvalues and creating a function which returns yes if the p-value of a feature is less than 0.05 or no if not. Therefore, all of our predicting variables are statistically significant with the threshold of p-value less than 0.05:

Table 3. Significance of independent variables

Features	**p-values**	**Is it statistically significant?**
Gender	0.001	Yes
Age	0.000	Yes
Disease	0.003	Yes
Service	0.000	Yes

RQ3. Which ML regression model predicts better LOS in a semi urban hospital?

The results of the Stacking Regressor predictions have proven the assumption of stability and lower variance as the stacking combines the performances of several algorithms [45]. As it is illustrated in table 4 below, the authors concluded that stacking regression performed better by having a MAE of 0.12, MSE of 0.14 and RMSE of 0.37. From the results in table 4, it can be seen clearly how the value of the mean square error for the stacked regression model is lower than for other models. This RMSE value is significantly lower than even the mean value of the patient length of stay variable, which was 2.979, and its standard deviation of 1.219. This means that the algorithm learned accurately from the used dataset and performed better. Therefore, on average, the LOS prediction of the regressive model was 0.37 units away from the real values.

Moreover, the coefficient of determination R² value of our model is 0.91. This means that 91% of the observed variability in the LOS in the hospital is captured and learned by the model and the remaining 9% is due to other factors, which can be for example the expertise of the treating physician, or even the climatic season during which the patient was admitted to the hospital.

Table 4. Model Evaluation Metrics

Model	**MAE**	**R-Squared**	**MSE**	**RMSE**	**Accuracy**
Random Forest Regressor	0.190	0.880	0.180	0.420	0.880
Extra Trees Regressor	0.130	0.890	0.160	0.400	0.890
Decision Tree Regressor	0.150	0.870	0.200	0.440	0.870
XGBoost Regressor	0.370	0.780	0.340	0.580	0.430
Deep Neural Network	0.430	0.620	0.580	0.760	0.780
Light GBM Regressor	0.350	0.800	0.310	0.560	0.800
SVR	0.450	0.610	0.590	0.770	0.610
Stacked Regressor	0.120	0.910	0.140	0.370	0.910

In Figure 9, the histogram illustrates how the Stacked Regression has a higher accuracy than all other models. Its accuracy is 91%. While Extreme Gradient Boost has the lowest accuracy, 43%. Concerning the R-Squared score, given that the closer its value is to 1, the better the model is, the author concluded that the Stacked model is able to make strong predictions about the length of stay at the hospital of a Patient given that the R-Squared stacked is 0.91. In addition, a very low Root Mean Squared Error was recorded in the Stacked model, followed by Extra Trees Regressor, Random Forest Regressor, Decision Tree Regressor, Light GBM, XGBoost, Deep Neural Network and SVR respectively.

From the MAE point of view, the Stacked model again won by presenting a value of 0.12, followed by Extra Trees Regressor, Decision Tree Regressor 0.15 and Random Forest Regressor 0.19. For the other four models, their MAE was over 0.30. In the end, for the MSE, in the first position is always stacking regression, which had the lowest MSE, 0.14. It was followed by Extra Trees Regressor with an MSE of 0.16, then Random Forest Regressor 0.18 and the Decision Tree Regressor with 0.2. The other four had an MSE of more than 0.3. In general, SVR, Deep Neural Network, Light GBM, and XGBoost did not perform well. It is clear that Extra Trees Regressor, Random Forest Regressor, and Decision Tree Regressor learned the structure of our Patient Length of Stay data well. Thus, by combining the different models, except for the Deep Neural Network, we were able to optimize our model to be able to generalize.

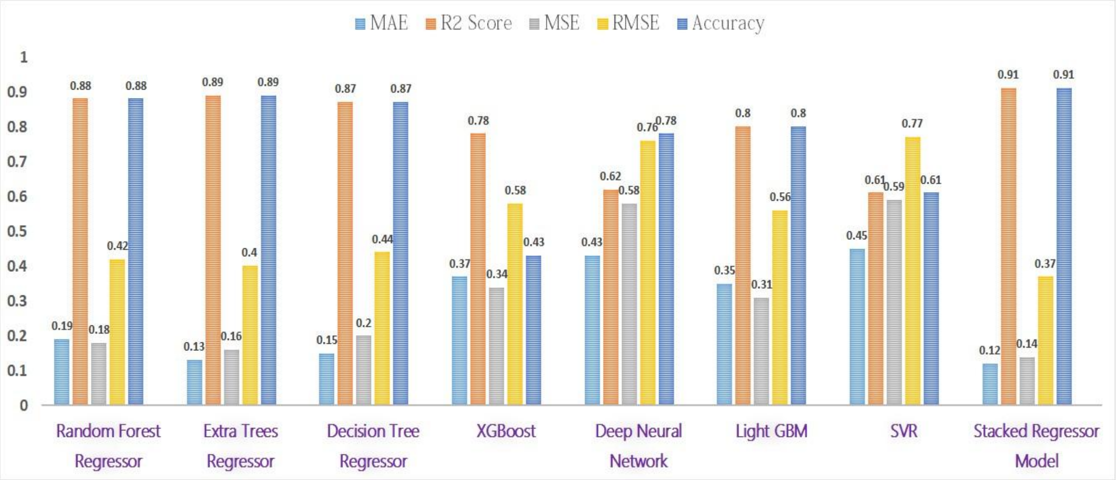


Figure 9. Model performances chart

In addition to the stacking model evaluation metrics, the authors were able to plot the correlation of the predicted values against the actual values using the test set. As can be seen below, the correlation between the two arrays (y predicted and actual y tests) is 0.95. On the right, there is the plot of residuals against the predicted values. In fact, residuals are understood as the actual value minus the predicted value. That right figure allowed seeing how well the constructed regressive model performed. Indeed, a regression model is efficient if the points representing the residuals are close to the horizontal line [46] knowing that residuals are plotted in axis Y and predicted values in axis X.

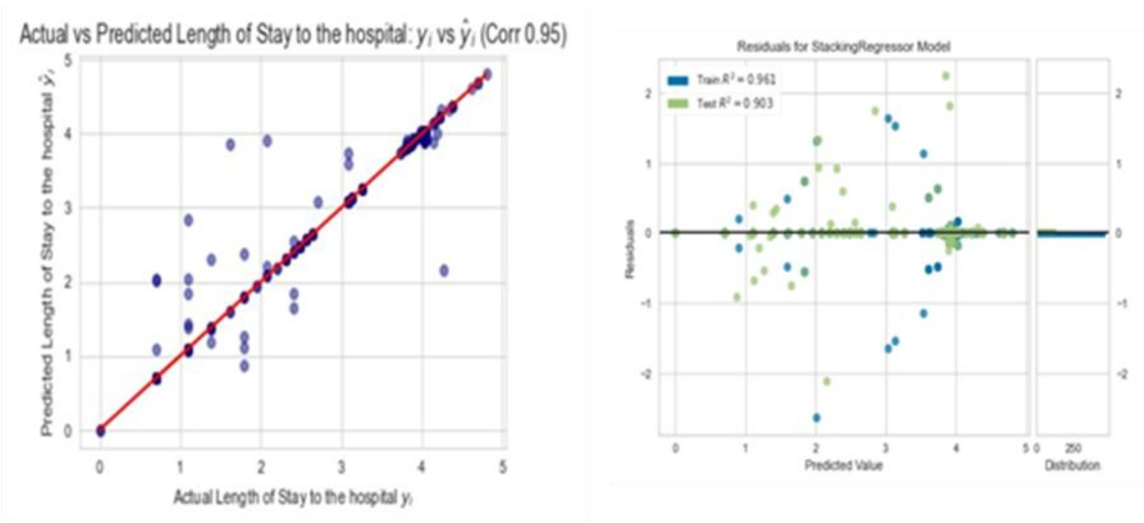


Figure 10. On the left the correlation between Actual vs Predicted LOS and on the right the residuals of the stacking model

The above right figure also confirmed the assumption of homoscedasticity as both axes have the same variance and they are moving together.

5. Conclusion AND FURTHER RESEARCH

This study has proved the necessity of using statistical analysis as a preliminary phase of a regression problem in Machine Learning. In fact, the first models that the authors developed before applying this preliminary phase had poor performances. When the authors applied the statistical analysis, they were able to remove skewness, outliers and check all the assumptions of the multi-linear regression, the performances of all the developed models were improved until reaching 91% and 0.91 as respectively the accuracy and the r-squared for stacking regression model for test set.

In the future, the author proposes to use different datasets for different hospitals to test with the same statistical and machine learning techniques used in this research to see if we can get a good performance like the one obtained here.

Acknowledgements

The authors would like to thank the Université de l'Assomption au Congo for its support.

References

[1] Ayesha Siddiqa, et al., (2021) "Robust Length of Stay Prediction Model for Indoor Patients", *Computers, Materials & Continua*, Vol.70, No.3, pp.5519-5536. https://doi.org//10.32604/cmc.2022.021666.

[2] S. Oliveira, et al., (2014) "Predictive Models for Hospital Bed Management using Data Mining Techniques", in R.Á., C.A., T.F., & S.K., New Perspectives in Information Systems and Technologies, Vol. 276, pp. 407-416, Cham: Springer. https://doi.org/10.1007/978-3-319-05948-8_39.

[3] B. Kavanaugh, J. Studeny, M.K. Cancilliere, K.A. & Holler, (2020) "Neurocognitive predictors of length of stay within a children's psychiatric inpatient program", *Child Neuropsychology*, Vol. 26, No. 1, pp.129-136. https://doi.org/10.1080/09297049.2019.1617843.

[4] S. Grampurohit& S. Sunkad, (2020) "Hospital Length of Stay Prediction using Regression Models", *2020 IEEE International Conference for Innovation in Technology (INOCON)*. https://doi.org/10.1109/INOCON50539.2020.9298294.

[5] M. Bassam, et al., (2021) "Prediction of COVID-19 Hospital Length of Stay and Risk of Death Using Artificial Intelligence-Based Modeling", *Frontiers in Medicine*, Vol. 8, 389. https://doi.org/10.3389/fmed.2021.592336.

[6] S. Sinha, Tushar, & S. Goel, (2021) "Research on Data Science Ensembles for Covid-19 Detection and Length of Stay Prediction", *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pp. 499-503. https://doi.org/10.1109/ICCCIS51004.2021.9397218.

[7] N. Kardani, A. Zhou, M. Nazem, & S.L. Shen, (2021) "Improved prediction of slope stability using a hybrid stacking ensemble method based on finite element analysis and field data", *Journal of Rock Mechanics and Geotechnical Engineering*, Vol. 13, pp. 188-201.

[8] H.N. Mpia, S.N. Mwendia, & L.W. Mburu, (2022) "Predicting Employability of Congolese Information Technology Graduates Using Contextual Factors: Towards Sustainable Employability", *Sustainability*, Vol. 14, No. 20, 13001. https://doi.org/10.3390/su142013001.

[9] B. Witkowska, & I. Frydrych, (2011) "Modelling the fabric tearing process", Soft Computing in Textile Engineering, Woodhead Publishing, pp. 424-489.

[10] M. Johnston, (2014) "Secondary Data Analysis: A Method of which the Time Has Come", *Qualitative and Quantitative Methods in Libraries*, Vol. 3, pp. 619-626.

[11] G. Bontempi, (2021) *Handbook Statistical foundations of machine learning,* ULB, (2nd ed.).

[12] A.I. Osman, et al., (2021) "Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia", *Ain Shams Engineering Journal*, Vo. 12, No. 2. https://doi.org/10.1016/j.asej.2020.11.011.

[13] A. Asselman, M. Khaldi, & S. Aammou, (2021) "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm", *Interactive Learning Environments*. https://doi.org/10.1080/10494820.2021.1928235.

[14] T. Chen, & C. Guestrin, (2016) "XGBoost: A Scalable Tree Boosting System", *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785-794. https://doi.org/10.1145/2939672.2939785.

[15] H.Y. Wu, et al., (2016) "Predicting postoperative vomiting among orthopedic patients receiving patient-controlled epidural analgesia using SVM and LR", *Scientific Reports*, Vol. 6, 27041.

[16] S. Samantaray, & A. Sahoo, (2021) "Prediction of suspended sediment concentration using hybrid SVM-WOA approaches", *Geocarto International*.

[17] B.T. Jijo, & A.M. Abdulazeez, (2021) "Classification Based on Decision Tree Algorithm for Machine Learning", *Journal of Applied Science and Technology Trends*, Vol. 2, No. 1, pp. 20-28.

[18] M. Shaheen, (2021) "Decision tree for PLOs of undergraduate computing program based on CLO of computer programming", *Interactive Learning Environments*.

[19] N. Mohapatra, K. Shreya, & A. Chinmay, (2020) "Optimization of the Random Forest Algorithm", in *Advances in Data Science and Management. Lecture Notes on Data Engineering and Communications Technologies,* Springer, Vol. 37, pp. 201-208.

[20] M. Bovo, et al., (2021) "Random Forest Modelling of Milk Yield of Dairy Cows under Heat Stress Conditions", *Animals*, Vol. 11, 1305.

[21] H. Adun, et al., (2021) "Novel Python-based "all-regressor model" application for photovoltaic plant-specific yield estimation and systematic analysis", Energy Sources, Part A: Recovery, Utilization, and Environmental Effects.

[22] S. Papadopoulos, et al., (2018) "Evaluation of tree-based ensemble learning algorithms for building energy performance estimation", *Journal of Building Performance Simulation*, Vol. 11, No. 3, pp. 322-332.

[23] F. Alzamzami, M. Hoda, & A.E. Saddik, (2016) "Light Gradient Boosting Machine for General Sentiment Classification on Short Texts: A Comparative Evaluation", *IEEE Access*, Vol. 4.

[24] M. Sokolov, & N. Herndon, (2021) "Predicting Malware Attacks using Machine Learning and AutoAI", *Proceedings of the 10th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2021)*, pp. 295-301.

[25] Y. Hou, et al., (2021) "Research on a novel data-driven aging estimation method for battery systems in real-world electric vehicles", *Advances in Mechanical Engineering*, Vol. 13, No.7, pp. 1-14.

[26] D. Chen, et al., (2020) "Deep Residual Learning for Nonlinear Regression", *Entropy*, Vol. 22, 193.

[27] R. Costache, et al., (2021) "Flash-flood potential index estimation using fuzzy logic combined with deep learning neural network, naïve Bayes, XGBoost and classification and regression tree", *Geocarto International*.

[28] A. Bashar, (2019) "Survey on evolving deep learning neural network architectures", *Journal of Artificial Intelligence and Capsule Networks*, Vol. 2, pp. 73-82.

[29] H.N. Mpia, & N.I. Baelani, (2021) "Gradient Back-Propagation Algorithm in the Multilayer Perceptron: Foundations and Case Study", *International Journal of Innovation and Applied Studies*, Vol. 32, No.2, pp. 271-290.

[30] A. Booeshaghi, & L. Pachter, (2021) "A SinaBooeshaghi, LiorPachter, Normalization of singlecell RNA-seq counts by log(x + 1) or log(1 + x)", *Bioinformatics*, Vol. 37, No. 15, pp. 2223-2224.

[31] M. Templ, J. Gussenbauer, & P. Filzmoser, (2020) "Evaluation of robust outlier detection methods for zero-inflated complex data", *Journal of Applied Statistics*, Vol. 47, No. 7, pp. 1144-1167.

[32] D. Turner, & H. Deng, (2020) "A Conceptual Introduction to Regression", *American Headache Society*, pp. 1047-1055.

[33] W. Wang, & Y. Lu, (2018) "Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model", *IOP Conference Series Materials Science and Engineering*, Vol. 324, No. 1, 012049.

[34] J. Qi, et al., (2020) “On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression”, *IEEE Signal Processing Letters*, Vol. 27, pp. 1485-1489.

[35] D. Chicco, M. Warrens, & G. Jurman, (2021) “The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation”, *PeerJ Computer Science*, Vol. 7, e623.

[36] H.J. Persson, & G. Ståhl, (2020) “Characterizing Uncertainty in Forest Remote Sensing Studies”, *Remote Sensing*, Vol. 12, No. 3, 505.

[37] A. Jierula, et al., (2021) “Study on Accuracy Metrics for Evaluating the Predictions of Damage Locations in Deep Piles Using Artificial Neural Networks with Acoustic Emission Data”, *Applied Sciences*, Vol. 11, 2314. https://doi.org/10.3390/app11052314.

[38] D. Bzdok, N. Altman, & M. Krzywinski, (2018) “Statistics versus Machine Learning”, *Nature Methods*, Vol. 15, No. 4, pp. 233-234.

[39] K. Rahul, et al., (2021) “Machine Learning Algorithms for Big Data Analytics”, in *Computational Methods and Data Engineering. Advances in Intelligent Systems and Computing*, Springer, Vol. 1227, pp. 359-367.

[40] M.O. Akinwande, H.G. Dikko, & A. Samson, (2015) “Variance Inflation Factor: As a Condition for the Inclusion of Suppressor Variable(s) in Regression Analyis”, *Open Journal of Statistics*, Vol. 5, No. 7, pp. 754-767.

[41] S. Uyanto, (2020) “Power Comparisons ofFive Most Commonly Used Autocorrelation Tests”, *Pakistan Journal of Statistics and Operation Research*, Vol. 16, No. 1, pp. 119-130.

[42] H. Elsayir, (2019) “Residual Analysis for Auto-Correlated Econometric Model”, *Open Journal of Statistics*, Vol. 9, pp. 48-61.

[43] E. Polat, & H. Ali, (2020) “Adaptive Reweighted Minimum Vector Variance Estimator of Covariance Used for as a New Robust Approach to Partial Least Squares Regression”, *Gazi University Journal of Science*, Vol. 33, No. 4, pp. 873-890.

[44] Q. Wang, et al., (2022) “Testing the intercept of a balanced predictive regression model”, *Entropy*, No. 11, 1594. https://doi.org/10.3390/e24111594.

[45] Y. Yao, et al., (2018) “Using Stacking to Average Bayesian Predictive Distributions (with Discussion)”, *Bayesian Analysis*, Vol. 13, No.3, pp. 917-1003.

[46] V. Basu, (2020) “Prediction of Stellar age with the help of Extra-Trees Regressor in Machine Learning”, *International Conference on Innovative Computing and Communication*.