Annals of Science and Technology - A, Vol 8 (2): 56-62, 2023
Copyright: An Official *Journal* of the Nigerian Young Academy
ISSN: 2 544 6320

sciendo

ARTICLE

Ensemble Algorithm for Simulated Corrosion Data-tentative

Chika O. Yinka-Banjo[1], Mary I. Akinyemi[2] and Blessing B. Yama[1]

[1]*Department of Computer Science, University of Lagos, Nigeria*
[2]*Department of Mathematics and Statistics, Austin Peay State University, Clarksville, 37044, Tennessee, USA*

Received 19th May, 2023, Accepted 18th September, 2023
DOI: 10.2478/ast-2023-0015

**Corresponding author*
Chika O. Yinka-Banjo (PhD), Email cyinkabanjo@unilag.edu.ng; Tel:: +2348033424289

Abstract

The application of robotics continues to advance and lends itself to various disciplines. Previous work has shown how Markov decision processes can be integrated in an algorithm to handle pipeline inspection. This has been extended to building models and addressing the errors in the model from the data collected. This paper made use of data collected by simulated robots in a pipeline environment, to build an ensemble. Five models were explored and analysed viz: Linear model, Generalised linear model, K-nearest neighbour, Neural network, and Random Forests models. The models were used to explore the responses acquired from the pipeline corrosion data. The performance of each of the ensembles are evaluated based on the Root Mean Square Error (RMSE), Adjusted R2 and Mean Absolute Error. Model evaluation show that although the averaging ensemble has a better fit (34.56%) than the stacking ensembles (17.22% and 16.98% respectively) considered, it also has the higher RMSE,(81.90) than the stacking models (40.91 and 38.53 respectively) forcing the decision maker to make a trade-off between fit and precision.
Keywords: Linear model, Generalised linear model, K-nearest neighbour, Neural network, Random Forests

BY-NC-ND © *Yinka-Banjo et al.* This work is licensed under the **Creative Commons Attribution-Non-Commercial-NoDerivs License 4.0**

1.0 Introduction

Pipeline installations are massive economic infrastructures and capital-intensive investments, so business owners must take adequate security measures to avoid major environmental consequences from the release of typically high volatile contents conveyed through them, which could occur as a result of bunkering, partial or total material failure. Environmental consequences have a high economic cost, and business owners prefer prevention to a partial or complete environmental collapse, which can take months or even years to reverse. Both environmental-dependent professions (e.g., fishing and farming) and pipeline owners face large financial losses. Oil leakage and bunkering cost fishermen and farmers billions of dollars each year owing to constant environmental deterioration or discharge of pipeline contents in Nigeria's Niger Delta region (Osuagwu and Olaifa, 2018). The recent return of restiveness in Nigeria's Niger Delta region has had disastrous economic ramifications for pipeline owners, who have had to spend billions of dollars to rebuild shattered pipe infrastructure. Militant organizations in the region have long complained that the region has been neglected, and that constant oil spills have robbed them of their environment-dependent employment. The government of the Federal Republic of Nigeria, in an effort to alleviate perceived concerns among the people of the region, launched an environmental clean-up procedure that will take many years to complete (Chijioke, 2016). Pipeline companies have used a variety of security and inspection methods in the past to protect their assets, including physical security and human security agents Kharrat et al., (2014). There have been obvious revelations about the limitations of these security approaches due to security lapses and fatigue in human security agents (Kim et al., 2015), (Lu et al., 2020). Pipeline business owners are now also heavily investing in research concerning new approaches to securing pipeline assets and are significantly interested in robotic automation (Ali and Choi, 2019). Robotic agents have been successfully used for pipeline monitoring, inspection, and repair. More so, there are conscious efforts in pipeline dependent industries to perfect the use of robotic agents to nearly replace all activities currently manned, because robotic agents are computers, and it will be wise to leverage on the excellent capabilities of computers for faster data processing, communication and better decision-making process. Reports from partial and full implementation of robotic agents for various pipeline dependent activities produced significant result and promise a more reliable and reactive approach (Abbas et al., 2018). Although robotic automation has been proven to be more cost effective, responsive and reliable for inspection, monitoring and repair of pipeline installations, unfortunately, pipeline dependent industries in Nigeria have not shown interest in partial or full implementation of robotic automation (Adegboye et al., 2019).

Pipeline inspection using an Unmanned Ground Vehicle (UGV) through Markov Decision Process (MDP) detected pipeline failures efficiently due to corrosion (Yinka-Banjo et al., 2021). UGV are capable of sensing and interacting with their surroundings, as well as working outside and across a range of terrain. The type of control system, method of movement, and planned operating region of UGV might all be used to categorize them. They are installed so that they may travel alongside the pipe for proper pipeline inspection (Kamel et al., 2020). To demonstrate how Markov Decision Process (MDP) could be used to manage pipeline inspection and substantiate the process as a cost effective and more reliable approach for pipeline inspection, five models were brought and utilized and later evaluated using some error metrics. The results reveal the size of corrosion and better guidance for maintenance and risk of accident.

2.0 Data Description

2.1 Description of the robots and how the data were generated
Data was collected through simulated robots and their environment. This was achieved by simulating a scenario of few surface-laid pipeline installations and one or more unmanned ground vehicles used for the inspection of pipelines at a refinery, such that defected parts are detected, marked, and recorded on the robot's memory or sent to a remote server in the refinery environment. The simulation which was done in Virtual Robot Experimentation Platform (VREP) made some assumptions on the UGVs having scanners, vision sensors and other assistive devices to allow them to make required contact with the environment. VREP has several simulated UGVs, for the purpose of this research, E-puck robot was used for the pipeline inspection. E-puck robot is an open hardware/software as well as the most recent mini mobile robot originally developed by the Swiss Federal Institute of Technology in Lausanne (EPEL) for the purpose of teaching. Simulated devices for epuck robot include, Integrated camera to gather information from surrounding environment, Infrared proximity sensors for acquisition of various useful data such as visual, audio, light intensity, speed, gravity and distance to object, Integrated Bluetooth to interact with a PC and can exchange information with neighbouring robots, Actuators for interaction with the environment, such as LEDs, motors and a speaker and Wired and wireless communication devices. Figure 1 show the features of the simulator used in collecting data from the simulated robots. VREP is versatile and has been used for complex simulation of robots due to the great functionality of the simulator. These functions are, Calculation of collision detection, Computer aided design data export or import, non-holonomic tasks for vehicles that are car-like and holonomic tasks in two to six dimensions (Path Planning), Simulation of proximity sensor, Camera-like sensor simulation, complete inverse and forward kinematics solver, Perfect for remote monitoring, hardware control, prototypes and verification, simulation of factory automation, etc.

The data consist of 200 reading points, comprising of 100 reading points from each of the 2 robots. Table 1 reports the summary statistics of the point readings and size of corrosion detected. Observe that the average corrosion size is 0.0320 and average readings for X, Y and Z points are 0.5561,0.7602 and 0.0667 respectively.

2.2 Detailed Justification of Density plots and box plots
The box plot provides a graphical representation of the five-number summary, gives an indication of the distribution of the data, and also shows the outlines present in the data. Here we have used the box plot to compare the distributions of the data from the 2 robots and also to examine their outliers. Identifying outliers is important for our work as outliers here need to be trimmed so that they do not bias our models.
Observe from the boxplot in Figure 2 that the corrosion readings from robot 2 have 2 outliers. The data was trimmed and a density plot (Figure 3) of the trimmed data was plotted. The density plot smoothens out the distribution of data values and reduces the noise in the data. Unlike the histogram, density plots are not affected by the number of bins. Thus, density plots are better at determining the distribution shape. The density plot here provides another view of our data after the outlines have been trimmed out.
The density plots in Figure 3 reveal the size of corrosion readings from both robots are right skewed with robot 2 still containing some outliers despite a 20% trim.

3. Method

3.1 Linear model (LM)

Linear models describe a continuous response variable as a function of one or more predictor variables. They help understand and predict the behavior of complex systems. Linear regression is a statistical method used to create a linear model. The model describes the relationship between a dependent variable, say yi (also called the response) as a function of one or more independent variables Xi (called the predictors). The general equation for a linear model is shown in Equation 1:

$$y=\beta 0+\sum[(\beta_i Xi+\varepsilon_i)] \quad (1)$$

where β represents linear parameter estimates to be computed and εi represents the error term
y_i is the dependent variable and X_i is the predictor variable

3.2 Generalised linear model (GLM)

GLMs are generalized linear models that allow for response variables with error distribution models other than a normal distribution (McCullagh and Nelder 2019). The GLM extends linear regression by enabling the linear model to be linked to the response variable through a link function and the size of each measurement's variance to be a function of its predicted value. Any GLM is made up of three parts:

1. Random Component: refers to the response variable's (Y) probability distribution, such as the normal distribution in linear regression or the binomial distribution in binary logistic regression.

2. Systematic Component: describes the explanatory variables in the model (X_1, X_2,..., Xk), more especially their linear combination in constructing the so-called linear predictor.,$\beta_0 + \beta_1X_1 + \beta_2X_2$ as in a linear regression with two dependent variables.

3. Link Function η or g(μ): specifies the link between random and systematic components. It says how the expected value of the response relates to the linear predictor of explanatory variables, e.g., $\eta = g(E(Y_i)) = E(Y_i)$ for linear regression, or $\eta = \text{logit}(\pi)$ for logistic regression.

3.3 K-nearest neighbour (KNN)

KNN is a straightforward algorithm that maintains all existing examples and categorizes new ones using a similarity metric (e.g., distance functions). The procedure involves categorizing a case based on a majority vote of its neighbours, with the case being allocated to the most frequent class among its K nearest neighbours, as determined by a distance function (Riskiawan et al., 2018). If K = 1, the case is automatically allocated to the class of its closest neighbour. Euclidean distance, hamming distance, Manhattan distance, and Minkowski distance are all popular distance functions utilized in KNN.

3.4 Neural network (NNET)

Neural networks (NNs) are massively parallel computers that maintain experimental information and allow for subsequent application. They create a simulation of the human brain in order to gather empirical information during the learning process. Knowledge is stored in inter neuronal connections (synapses) (Zhang, 2020). NNs can generalize their learnt knowledge, which is in addition to their ability to learn. An artificial neural network is made up of a set of neural nodes connected by weighted nodes. Every node can represent a creature's neuron, and the link between them is the same as the synaptic connection between neurons. Input layers, hidden layers, and output layers are the three levels of units in the most popular form of neural network (Ognjanovski, 2019). It's known as a multilayer perceptron (MLP). A layer of input units is linked to a hidden unit layer, which is linked to a layer of output units.

3.5 Random Forests (RF)

Breiman created RF, a classification strategy based on the usage of classification and regression trees, in 2001. (Fawagreh et al., 2014). By recursive binary partitioning into areas that are progressively homogenous with regard to the class variable, classification trees create rules for allocating current data into classes using numerical and/or categorical predictor variables. The nodes represent the homogenous areas. RF fits several classification trees to a data set and then aggregates all the trees' predictions. The procedure starts by selecting many bootstrap samples from the data (for example, 500). Out-of-bag observations are observations in the original data set that do not appear in a bootstrap sample. Each bootstrap sample is fitted with a classification tree, and a limited number of randomly selected variables are accessible for binary splitting at each node. The trees have reached full maturity and are utilized to anticipate the out-of-bag observations. An observation's projected class is determined by a majority vote of the out-of-bag forecasts for that observation, with ties broken at random. Since the ensemble's basic constituents are tree-structured predictors, and each of these trees is generated using an injection of randomness, Ishwaran et al. (2008) named this approach random forest.

The target here is to get the best model that can be fed back into the UGVs for forecasting and predictions. The UGVs had been used to collect data via the simulator VREP. The data collected was used to build the selected models.

3.6 Model performance evaluation

We measure model prediction performance using:

• Root Mean Square Error (RMSE): RMSE is an absolute measure of fit. It is computed as shown in Equation 2

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

The lower the RMSE is, the better the fit of the model (Chai and Draxler, 2014). The RMSE like any statistical measure, has its strengths and weaknesses. Some of the strengths of the RMSE include that the interpretation is simple and easy to follow. Since it is an absolute measure of the average distance that the data points fall from the predicted values using the units of the dependent variable. It can assess prediction precision directly. Furthermore, The root mean square error is a standard metric in many fields. Analysts frequently use it to evaluate model performance.

The drawbacks of the RMSE include that it is sensitive to Outliers. The squaring process in the root mean square error causes it to give a disproportionately higher weight to larger errors, making it more sensitive to outliers. It is also sensitive to overfitting, The root mean square error decreases every time you add an independent variable to the model, even when it's just a chance correlation between variables. It never increases when adding a variable to the model. Because you want a small RMSE, you can create the appearance of a better model simply by adding more variables, even when they don't make sense. Finally, the RMSE is sensitive to the scale of the dependent variable. So interpreting it requires additional knowledge about the dependent variable and its scale. In some cases, the root mean square error might not be comparable across different datasets or units of measurement.

- Adjusted R2: Adjusted R2 is a relative measure of fit whose values range from 0 to 1. It is computed as shown in Equation 3

$$\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \underline{y}_i)^2} \qquad (3)$$

The closer the Adjusted R2 value is to 1 the better the model fit.

A major pro of the Adjusted R2 is that it uses a standardized scale (0 – 100%) to evaluate the goodness-of-fit of the model making it easier to compare Adjusted R2 between different studies and even entirely different outcome variables.

However, the Adjusted R2 cannot assess precision because it is a unitless relative measure. It also subject to the same outlier and overfitting cons of the RMSE.

- Mean Absolute Error (MAE): MAE is a metric that estimates the non-directional average magnitude of a group of forecasts' mistakes (Chai and Draxler, 2014). It averages the absolute differences between forecast and actual observation over the test sample, with all individual deviations given equal weight. It's calculated as shown in Equation 4

$$\sum_{i=1}^{n}|y_{i-}\hat{y}_i| \qquad (4)$$

An advantage of the MAE is that since we are taking the absolute value, all of the errors will be weighted on the same linear scale. putting less weight on outliers and our loss function provides a generic and even measure of how well our model is performing.

The drawback of this metric is that the large errors coming from the outliers end up being weighted the exact same as lower errors. This

might result in the model being great most of the time, but making a few very poor predictions every so-often.

The trade-off between model interpretability and accuracy is an important consideration in the development of machine learning models. Interpretability refers to the ability to understand and explain how a machine learning model makes decisions.

Accuracy on the other hand refers to the ability of a model to accurately predict outcomes on a given task. In many cases, achieving both interpretability and accuracy is not always possible, and a trade-off must be made between the two. In these cases, the choice between interpretability and accuracy will depend on the specific application and the goals of the model. The trade-off between interpretability and accuracy can however be balanced by designing models that prioritize both interpretability and accuracy viz ensemble models.

4.0 Results and Discussion

The data was collected in excel and then exported to R (R Core team, 2021) for analysis.

LM, GLM, KNN, NNET and RF models were built and the performance of each of the models were evaluated with the RMSE, Adjusted R2 and MAE described in Section 3.6. Figure 4 presents the 95% confidence intervals (CI) for each of RMSE, Adjusted R2 and MAE for each of the models considered. The (CI) measures the margin of uncertainty around the point estimate. The best estimation for the parameter value is the point estimate. When utilizing a sample to estimate a whole population, the margin of error compensates for the uncertainty involved. The accuracy is revealed by the breadth of the CI around the point estimate. The margin of error is limited if the range is tight, and there is only a small range of feasible values indicative of an accurate estimate. If the interval is vast, however, the margin of error is substantial, and the actual parameter value is likely to lie somewhere within that larger range. representing a rough estimation. Observe from Figure 4 that RMSE for all the models have a wide range while the MAE all have a smaller range when compared to the RMSE. The CI for the R2 for all the models is small while that of the KNN model is wide. Hence the MAE and Adjusted R2 estimates are more precise than the RMSE estimates. However, the CI for Adjusted R2 for the KNN model indicates that the estimate is not as precise as the others. Figure 5 presents the scatter plot matrix showing the relationship between the MAE for pairs of the models considered.

Table 2 represents the correlation matrix between pairs of model results, we observe that all the results have high correlation making them poor candidates for pairs of ensemble combinations. Hence rather than consider pairs of models, ensembles of all five models were built at once. The model evaluation results are presented in Table 3. It can be observed from Table 3 that although the stacking ensembles using both LM and GLM have the lowest RMSE, the random forest model has the highest Adjusted R2 indicating it gives a better fit than the other models. The RMSE value here tells us the average deviation between the predicted values made by the model and the actual values.

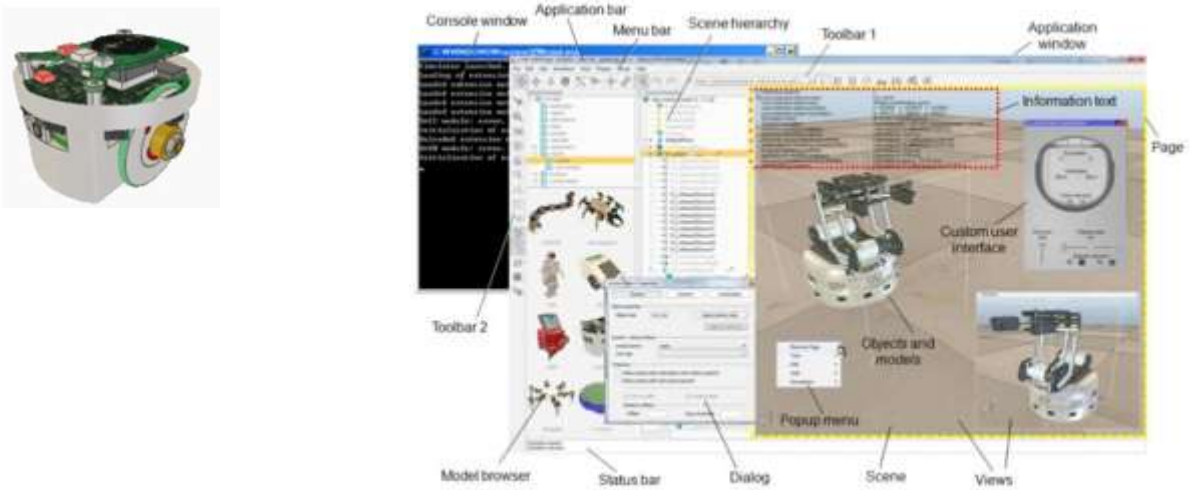

Figure 1: User interface of the VREP Simulator used and E-puck robot used Source: Tharin, 2016)

Table 1: Summary Statistics of Readings

Reading point and Size of corrosion	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
X	0.0137	0.0508	0.4717	0.5561	0.9595	0.9961
Y	0.5391	0.7109	0.7988	0.7602	0.8086	0.9238
Z	0.0039	0.0273	0.0664	0.0667	0.0898	0.1929
Size of Corrosion	0.0001	0.0022	0.0133	0.0957	0.0320	6.1035

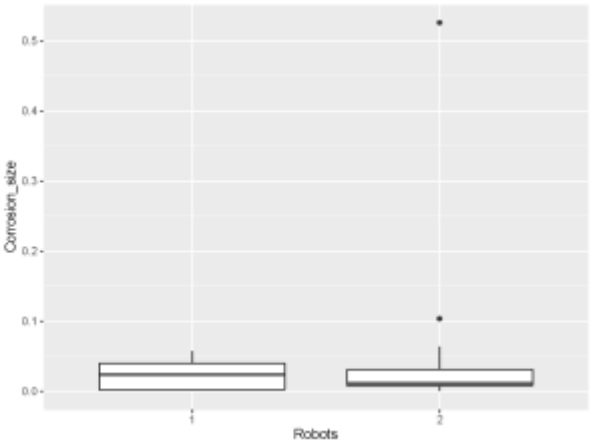

Figure 2: Boxplot of corrosion size by Robot

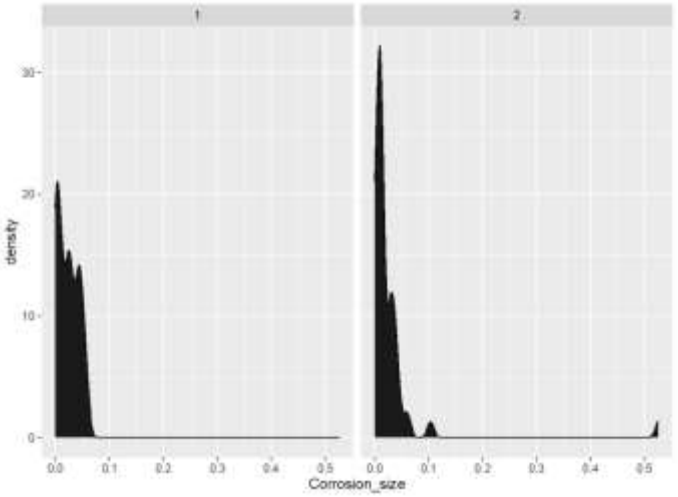

Figure 3: Density plot of corrosion size detected by Robot 1 and 2

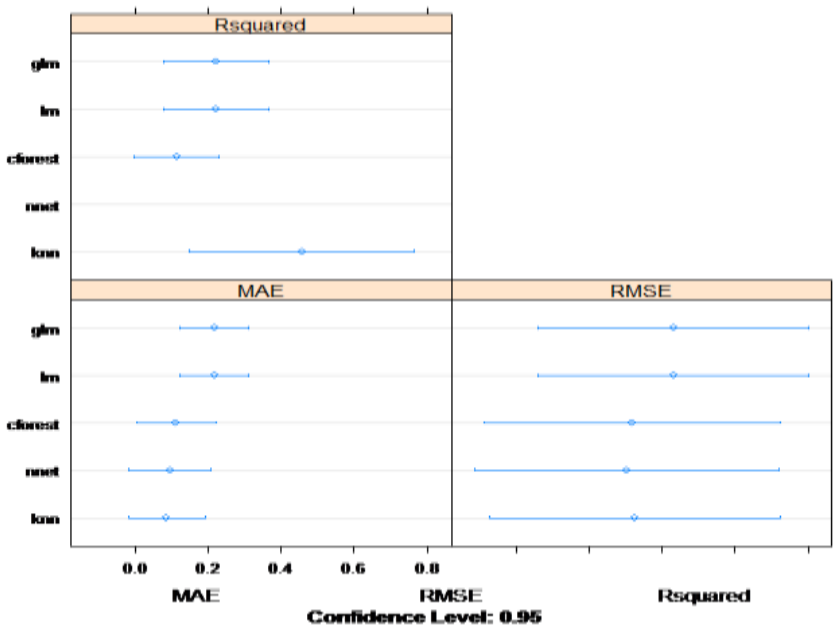

Figure 4: 95% Confidence interval for R2, MAE and RMSE

Table 2: Correlation Matrix between

	KNN	LM	GLM	NNET	R-Forest
KNN		0.9910	0.9910	0.9928	0.9952
LM	0.9910			0.9892	0.9917
GLM	0.9910			0.9892	0.9917
NNET	0.9928	0.9892	0.9892		0.9994
R-Forest	0.9952	0.9917	0.9917	0.9994	

Table 3: Model diagnostics for Ensembles

Models	RMSE	Adjusted R^2	MAE
Linear model	0.9190	0.0061	0.2854
Generalized Linear model	0.9190	0.0061	0.2854
Random forest	0.5245	0.9984	0.0780
K-nearest neighbour	0.8771	0.0073	0.1311
Neural network	0.8813	0.0025	0.1446
Averaging ensemble	0.8190	0.3456	0.1786
Stacking using LM	0.4091	0.1722	0.1682

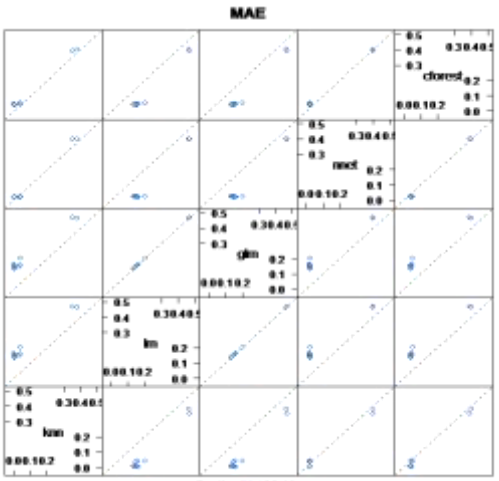

Figure 5: Scatter Plot Matrix of Correlation Between Pairs of Models

The R2 value tells us what percentage of the variation in the dependent variable the predictor variables in the model are able to explain. e.g. For the average ensemble model, the RMSE value tells us that the average deviation between the predicted pipeline corrosion data and the actual pipeline corrosion data is 0.8190. The adjusted R2 value tells us that the predictor variables in the average ensemble model are able to explain 34.56% of the variation in the pipeline corrosion data.

Ensemble Averaging is used here to improve model performance by introducing diversity to reduce variance. Here, each ensemble member contributes equally by making a vote using their predictions. This introduces a diversity in some sense which translates to a reduced variance and hence improve the ability to generalize beyond the training data, giving a better model fit.
Stacking joins the machine learning models, by combining the output of many models. The performance improvement when stacking is done is often not too far from that of the best model.

In our case, although the averaging ensemble has a better fit than the stacking ensembles considered, it also has the highest RMSE, forcing the decision maker to make a trade-off between fit and precision.
The trade-off between adjusted R2 and RMSE depends on the context and the goal of the modelling. In some cases, having a high adjusted R2 and a low RMSE may be more important than having a high number of precise predictions, such as in cases where the goal is to minimize error and maximize the ability of the model to explain the variance in the data. However, in other cases, having a high number of precise predictions may be more important, such as in cases where the goal is to make accurate predictions or classifications. In our case, we are keen on being able to precisely predict pipeline corrosion as this will help decision makers proactively plan maintenance exercises on pipelines.

5.0 Conclusion

In this research, ensemble method which takes several models into account, in this case, LM, GLM, KNN, NNET and RF has been utilized. The choice of ensemble methods is largely driven by simplicity, ease of computation and interpretations as well as improved model performance. Model performances were evaluated with RMSE, Adjusted R2 and MAE values. It can be observed that although the stacking ensembles using both LM and GLM have the lowest RMSE, the random forest model gives the best fit when compared to the other models.

Furthermore, although the averaging ensemble has a better fit than the stacking ensembles considered, it also has the highest RMSE, forcing the decision maker to make a trade-off between fit and precision.
There are several other factors to support these findings. Because it depends on the sniffing abilities of cameras and sensors, the UGV has a better chance of identifying leaks or rust in pipelines than physical eyes. Corrosion can be so minute that human eyes miss it, but a UGV will typically detect it, allowing the maintenance personnel to be more proactive rather than waiting until substantial damage has occurred as a result of ignored corrosion. When compared to human inspectors, UGVs are inherently reusable and cost effective.
In conclusion, the UGVs collected data via the simulator, VREP and the data was used to build models and the target is to get the best model that can be fed back into the UGV for forecasting and predictions. The best model was found to be averaging ensemble as it provides the most balance between fit and precision. This model will be fed back in our future work.

Conflict of Interest

There is no conflict of interest whatsoever.

Author Contribution

Conception: [CYB, MIA]
Design: [CYB, MIA]
Execution: [CYB, MIA, BOY]
Interpretation: [CYB, MIA]
Writing the paper: [CYB, MIA, BOY]

References

Abbas, M.Z., Abu Baker, K., Ayaz, M., Mohamed, H., Tariq, M., Ahmed, A. and Faheem, M. (2018). Key factors involved in pipeline monitoring techniques using robots and WSNs: Comprehensive survey. Journal of Pipeline Systems Engineering and Practice, 9(2), p.04018001.

Adegboye, M.A., Fung, W.K. and Karnik, A. (2019). Recent advances in pipeline monitoring and oil leakage detection technologies: Principles and approaches. Sensors, 19(11), p.2548.

Ali, H. and Choi, J.H. (2019). A review of underground pipeline leakage and sinkhole monitoring methods based on wireless sensor networking. Sustainability, 11(15), p.4007.

Breiman, L. (2001). Random forests. Machine learning, 45(1), pp.5-32.

Chai, T. and Draxler, R.R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)–Arguments against avoiding RMSE in the literature. Geoscientific model development, 7(3), pp.1247-1250.

Chijioke, A.N, (2016). NetNaiga. Retrieved September 12, 2016, from https://www.netnaija.com/forum/general/news/25353-buhari-set-visit-niger-delta-june-2-flag-clean-ogoni-land

Fawagreh, K., Gaber, M.M. and Elyan, E. (2014). Random forests: from early developments to recent advancements. Systems Science & Control Engineering: An Open Access Journal, 2(1), pp.602-609.

Ishwaran, H., Kogalur, U.B., Blackstone, E.H. and Lauer, M.S. (2008). Random survival forests. The annals of applied statistics, 2(3), pp.841-860.

Kamel, M.A., Yu, X. and Zhang, Y. (2020). Formation control and coordination of multiple unmanned ground vehicles in normal and faulty situations: A review. Annual reviews in control, 49, pp.128-144.

Kharrat, M., Ichchou, M.N., Bareille, O. and Zhou, W. (2014). Pipeline inspection using a torsional guided-waves inspection system. Part 2: Defect sizing by the wave finite element method. International Journal of Applied Mechanics, 6(04), p.1450035.

Kim, J.H., Sharma, G., Boudriga, N., Iyengar, S.S. and Prabakar, N. (2015). Autonomous pipeline monitoring and maintenance system: a RFID-based approach. EURASIP Journal on Wireless Communications and Networking, 2015(1), pp.1-21.

Lu, H., Iseley, T., Behbahani, S. and Fu, L. (2020). Leakage detection techniques for oil and gas pipelines: State-of-the-art. Tunnelling and Underground Space Technology, 98, p.103249.

McCullagh, P. and Nelder, J.A. (2019). Generalized linear models. Routledge.

Ognjanovski, G., (2019). Everything you need to know about neural networks and backpropagation—machine learning easy and fun. Towards Data Science.

Osuagwu, E.S. and Olaifa, E. (2018). Effects of oil spills on fish production in the Niger Delta. PloS one, 13(10), p.e0205114.
R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.URL https://www.R-project.org/.

Riskiawan, H.Y., Puspitasari, T.D., Hasanah, F.I., Wahyono, N.D. and Kurnianto, M.F. (2018). Identifying Cocoa ripeness using K-Nearest Neighbor (KNN) Method. In 2018 International Conference on Applied Science and Technology (iCAST) (pp. 354-357). IEEE.

Tharin, J. (2016). Virtual Robot Experimentation Platform. Retrieved May 18, 2016,from http://www.coppeliarobotics.com/helpFiles/en/welcome.htm.

Yinka-Banjo, C.O., Akinyemi, M.I., Nwadike, C.O., Misra, S., Oluranti J., Damasevicius, R. (2021). Unmanned Vehicle Model Through Markov Decision Process for Pipeline Inspection. In: Tripathi, M., Upadhyaya S. (eds), ICDLAIR 2019. Lecture Notes in Networks and Systems, vol 175. Springer, Cham. https://doi.org/10.1007/978-3-030-67187-7_33

Zhang, X.D. (2020) Neural Networks. In: A Matrix Algebra Approach to Artificial Intelligence. Springer, Singapore.
https://doi.org/10.1007/978-981-15-2770-8_7