MACHINE LEARNING BASED HEALTH PREDICTION SYSTEM USING IBM CLOUD AS PAAS

[1]Shifa Mahemood Shaikh, [2]Sujata. A. Gaikwad (Prof)
[1,2]TPCT'S College of Engineering, Dharashiv
[1]shifashaikh544@gmail.com

Abstract: Designing a system that emphasizes adaptability and prompt response is essential to bridge existing gaps in healthcare services. The main objective is to enhance patient care and strengthen the quality of services delivered by hospitals. By leveraging Machine Learning (ML) and Cloud-based solutions (PaaS), patient health can be monitored in real time, especially for those in critical conditions. This approach prioritizes improved decision-making and continuous supervision in medical environments. To reduce expenses, the IBM Cloud is locally emulated within the system. The proposed model integrates multiple algorithms, including Naïve Bayes, Logistic Regression, and Decision Tree Classifier, to build a robust ensemble framework for predicting critical health conditions. Furthermore, a mobile application called the 'Critical Patient Management System (CPMS)' has been developed, enabling healthcare providers to track patient conditions remotely and in real time. The application equips doctors with effective tools to monitor critical patients even in their absence, ensuring reliable healthcare management.

"Index terms – *Machine Learning in Healthcare, Predictive Analytics, Cloud-based Health Monitoring, Logistic Regression, Ensemble learning Models".*

1. INTRODUCTION

Critical patient monitoring systems allow doctors to observe and manage multiple patients simultaneously, even from remote locations. Such systems also help physicians determine appropriate medication doses for each patient [1]. These technologies can significantly simplify the development and testing of ICU decision-support mechanisms. Devices like vital sign monitors, ventilators, dialysis units, and other clinical tools are commonly employed to support critically ill patients while their bodies recover. Traditionally, these machines are operated manually, relying heavily on continuous supervision and diagnostic reports. To overcome these limitations, our approach utilizes modern technology—particularly auto-deployable machine learning models combined with cloud computing—to make patient monitoring more efficient and less dependent on manual processes.

ML algorithms can forecast short-term changes in a patient's condition, such as improvement, deterioration, or the need for urgent medical attention. To maximize effectiveness, IBM Cloud was selected as the PaaS platform, given its compatibility with public, private, and hybrid deployment models [2]. Before launching, our system required IBM Cloud and IBM Watson Studio to manage storage, testing, and deployment. By running models in the cloud, the system can continually learn from automatically deployed data. The proposed 'Critical Patient Management System (CPMS)' can also connect with Bluemix services [3]. In addition, the growing reliance on digital health solutions worldwide highlights the importance of adopting scalable, cloud-driven systems in developing countries. By combining AI-powered predictions with accessible mobile platforms, healthcare delivery can be made more proactive and patient-centric.

A key contribution of this work is the integration of auto-deployable ML models into cloud storage, along with methods for parameter tuning and algorithm optimization. The healthcare sector, particularly in countries such as Bangladesh, has historically underutilized such technologies [4]. Unlike other industries that benefit from digital innovations, healthcare often struggles with outdated practices. This gap results in severe consequences—patients in emergencies may suffer permanent harm or death due to delayed monitoring and response [5]. Moreover, when doctors are not present, communication often relies solely on phone calls, which can create misunderstandings.

Our research proposes a system where physicians can remotely monitor vital patient signs using ML models, ensuring more intelligent care. Cloud computing enables access to real-time patient data from anywhere, allowing doctors to oversee multiple patients efficiently. Families also benefit by receiving regular updates on patient health without needing to visit hospitals frequently.

2. LITERATURE SURVEY

This study examines approaches to building a decision tree classifier [18] in situations where data is divided into two vertical segments. Here, Alice possesses one portion while Bob owns the other. Both parties aim to collaboratively construct a decision tree classifier without disclosing their private data, as this could lead to security risks. A protocol is proposed that enables the classifier to be developed while maintaining privacy. This solution, which relies on the scalar product protocol, incorporates an untrusted third-party service. Compared to existing methods, our implementation of the scalar product protocol performs more efficiently [32]. Recent studies have emphasized the integration of explainable AI (XAI) in healthcare prediction models, allowing doctors to interpret model decisions with greater transparency.

To evaluate ensemble modeling techniques, experiments were conducted using the well-known WEKA data mining platform. Six algorithms—including neural networks, decision trees [18, 32], support vector machines, linear regression, and others—were applied to real-world datasets, such as property transaction records. Performance was analyzed through the non-parametric Wilcoxon Signed-Rank Test to compare results with baseline models. Findings revealed that no single algorithm consistently produced the best ensembles. Instead, hybrid multi-model strategies demonstrated the greatest potential. Key terms: ensemble methods, bagging, boosting, stacking, property valuation [29].

The Naïve Bayes classifier simplifies learning by assuming feature independence. While this assumption is not always accurate, Naïve Bayes frequently achieves results comparable to more complex methods. The main focus of related work is to determine which data characteristics influence its effectiveness [25]. Monte Carlo simulations were employed to systematically study classification accuracy across diverse synthetic problem sets. Results showed that Naïve Bayes achieves optimal performance when features exhibit low-entropy distributions. Furthermore, Naïve Bayes adapts well to both independent and dependent functional relationships. Interestingly, classification accuracy is less influenced by feature dependency (as measured by mutual information) than by the amount of class-related information lost due to independence assumptions.

In ML workflows, hyperparameters, normalization rules, and optimization settings must be carefully selected [16, 27, 30, 32]. Unfortunately, this process is often viewed as a 'black art,' learned primarily through trial and error. To address this, automated feature engineering and Bayesian optimization have been introduced. Here, generalized learning performance is modeled using Gaussian processes (GPs), which generate posterior distributions from prior experiment data. This guides the selection of future parameters, enhancing performance beyond manual tuning. Experiments confirm that GPs and related inference methods significantly influence the success of optimization. Advanced strategies, such as multi-kernel parallel processing that accounts for varying training times, outperform earlier methods and even rival expert-level tuning.

Another time-consuming task in computer vision involves nearest-neighbor searches in high-dimensional data. Exact solutions faster than linear search are not currently known, though approximate algorithms can deliver major speed improvements with minimal accuracy loss. However, choosing the right algorithm and its parameters for a specific dataset remains a challenge. This study [28] addresses this by introducing a system that automatically identifies the optimal method and parameter configuration based on dataset characteristics and desired accuracy. Hierarchical k-means trees, among other techniques, were shown to perform exceptionally well. Trials confirmed that randomized K-D trees work best for certain datasets [18, 29, 32]. To support this, a publicly available software library was released, offering query times nearly ten times faster than existing solutions while automatically adjusting parameters.

3. METHODOLOGY

i) Proposed Work:

A modern Health Prediction Framework based on Machine Learning [16, 27, 30, 32] is proposed in this research. The system is designed to run on a Cloud Platform-as-a-Service (PaaS). By employing algorithms such as Naïve Bayes, Logistic Regression, and others, it enables real-time training and deployment of predictive models for accurate healthcare analysis. To make the solution practical, a mobile application called Critical Patient Management System (CPMS) has been developed. This app allows doctors and nurses to instantly view patient data in real time.

The system emphasizes the delivery of smart healthcare solutions by ensuring proactive interventions and immediate feedback within hospitals. Through advanced ML models, it generates real-time predictions that provide healthcare staff with actionable insights into patient conditions. The integration of multiple models—like Naïve Bayes and Logistic Regression—ensures comprehensive data evaluation. This layered approach increases the reliability and accuracy of predictions, capturing multiple aspects of patient health [3–17].

The CPMS mobile app offers an intuitive interface that enables healthcare professionals to easily interact with patient data, monitor critical health indicators, and receive system-driven recommendations. This significantly enhances efficiency in healthcare management.

ii) System Architecture:

The proposed CPMS architecture incorporates a mobile application that serves as the entry point for doctors to record patient vitals. The data is processed on a locally simulated IBM Cloud (PaaS), which hosts and executes multiple ML algorithms (Naïve Bayes, Logistic Regression, Decision Tree Classifier). These algorithms are used to predict critical health risks in real time.

Once processed, the app immediately provides doctors with predictive outcomes, enabling them to make timely decisions. This cloud-driven and scalable structure ensures that medical professionals can remotely monitor patients from different geographical locations without delays.

The architecture leverages the cloud's elasticity, ensuring reliable data storage, rapid processing, and secure handling of sensitive patient records. This design guarantees both flexibility and scalability while supporting continuous monitoring of multiple patients simultaneously.

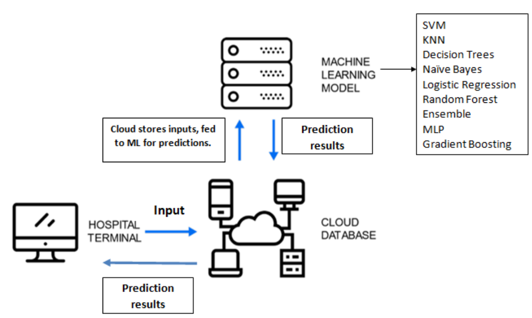


Fig 1 Proposed Architecture

iii) Data collection:

The dataset used for training the ML models includes fifteen attributes representing different health indicators. These features consist of age, gender, chest pain type (CP), resting blood pressure (TRESTBPS), cholesterol (CHOL), fasting blood sugar (FBS), resting ECG (RESTECG), maximum heart rate achieved (Thalach), old peak, slope, and thalassemia type (THAL), along with a class label that categorizes a patient's condition as either normal or abnormal (1). This dataset enables effective training and validation of predictive models.

The initial stage involves capturing a patient's vital parameters such as temperature, blood pressure, and other physiological readings. These values may be manually entered by medical staff or collected through sensor-equipped devices.

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	class
63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
57	1	0	140	192	0	1	148	0	0.4	1	0	1	1
56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
44	1	1	120	263	0	1	173	0	0	2	0	3	1
52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
57	1	2	150	168	0	1	174	0	1.6	2	0	2	1

Healthcare providers use the CPMS mobile app [3] to input and upload vital signs. The application serves as a user-friendly interface that seamlessly connects with the cloud system. The recorded information is transmitted to the locally simulated IBM Cloud (PaaS) [17], where ML models analyze the inputs.

Each algorithm—including Naïve Bayes, Logistic Regression, and Decision Tree Classifier—processes the data individually to deliver predictive insights. The system then provides real-time feedback through the CPMS app, displaying alerts, predictions, and decision-support recommendations.

Past patient records are securely maintained in the cloud database. This archival data not only ensures reliable long-term storage but also supports continuous improvement of the ML models as they retrain on accumulated datasets.

The architecture is designed for scalability, enabling smooth handling of larger datasets as patient numbers increase. Remote access functionality further allows healthcare professionals to monitor patients from different locations, ensuring flexibility and convenience.

iv) Algorithms:

"Support Vector Machine (SVM)": Chosen for its strong classification abilities, SVM identifies the optimal boundary that separates patient health states. It works effectively with complex, high-dimensional data, improving the accuracy of predictions in CPMS.

"K-Nearest Neighbours (KNN)": This algorithm predicts patient conditions by comparing them to the states of nearby data points. Each patient's vitals are treated as a data instance, and the algorithm assigns outcomes based on the closest neighbors. As part of the ensemble, KNN enhances adaptive and real-time decision-making.

Decision Tree: By modeling patient health using a tree-like decision structure, this algorithm makes outcomes transparent and easy to interpret. Its visual nature helps medical professionals understand the reasoning behind predictions, a critical factor for healthcare applications [18].

Naïve Bayes: : This simple yet effective probabilistic model estimates the likelihood of abnormal conditions based on vital sign data. Despite assuming feature independence, it consistently delivers reliable predictions, strengthening the ensemble model [16].

Logistic regression: Specifically designed for binary classification, this model predicts whether a patient's condition is stable or critical. It provides clear probability estimates that are practical for real-time clinical decision-making [17].

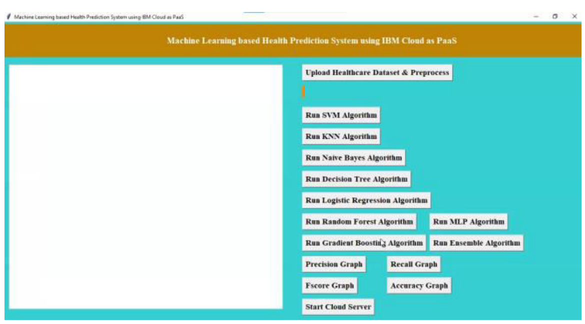
Random forest: By training multiple decision trees on random subsets of data, Random Forest minimizes overfitting and improves prediction stability. Its ensemble nature makes it ideal for analyzing large, complex patient datasets [20].

"Ensemble Algorithm: The system integrates SVM, Random Forest, and Decision Tree classifiers into a hybrid ensemble framework. SVM identifies intricate patterns, RF enhances robustness, and DT ensures interpretability—together they maximize prediction accuracy.

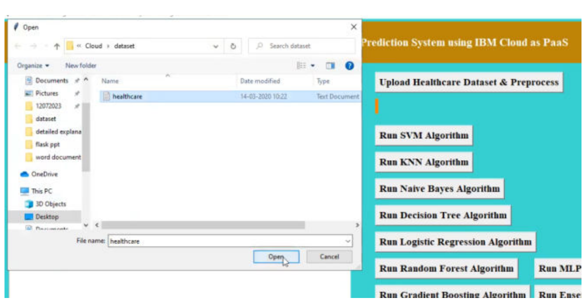
Multilayer Perceptron (MPL): As a type of neural network, MLP captures nonlinear patterns in patient health data. Its inclusion boosts the performance of the ensemble, particularly in complex prediction tasks [19].

Gradient Boosting: This iterative ensemble method enhances predictive strength by progressively correcting prior errors. It produces highly accurate health assessments, making it well-suited for real-time monitoring in CPMS.

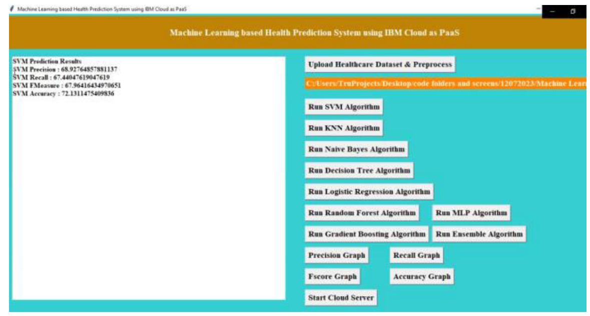
4. EXPERIMENTAL RESULTS



"Fig 3 Home screen"

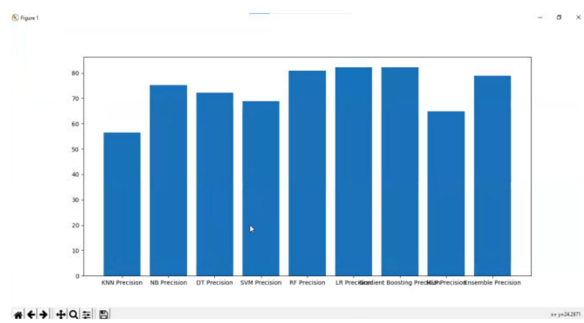


"Fig 4 Upload healthcare dataset and preprocess"



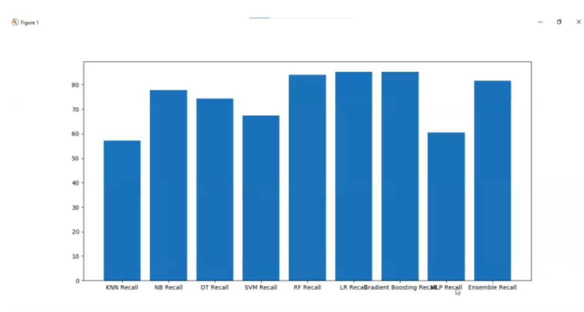
"Fig 5 Run Algorithm"

Precision checks how accurate positive predictions are by showing how many of those statements turned out to be right.



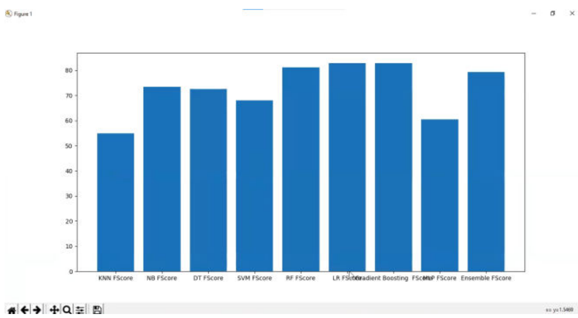
"Fig 6 Precision graph"

Recall measures how all the relevant examples and shows how many real positives were correctly predicted.



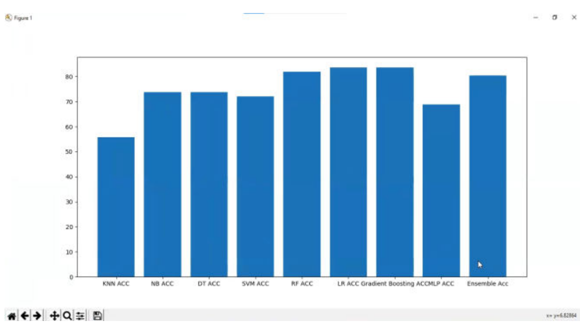
"Fig 7 Recall graph"

The role of F1 score is that it combines precision and recall into a single metric that measures how accurate and full predictions are.

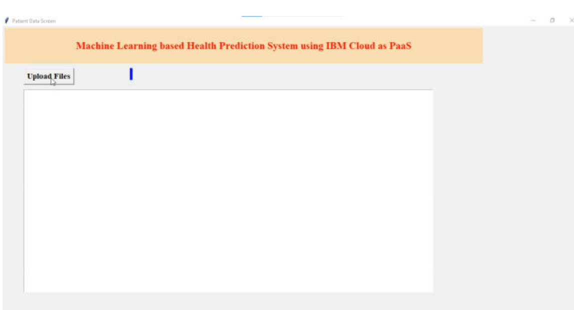


"Fig 8 Fscore graph"

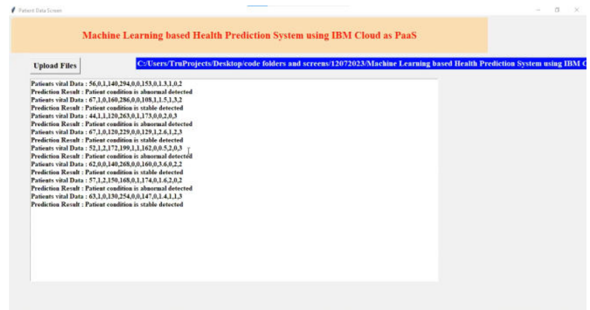
Accuracy shows the percentage of properly classified instances and it also measures how accurate predictions are generally.



"Fig 9 Accuracy graph"



"Fig 10 Start cloud server"



"Fig 11 Upload files"

5. CONCLUSION

This project demonstrates the effective integration of cloud technology with machine learning [16], creating a more seamless process for detecting abnormal health conditions. By combining multiple algorithms, the system delivers holistic predictions that account for various health factors. The use of a locally simulated cloud ensures that testing can be performed thoroughly without requiring significant financial resources, making it practical for students and developers.

Through the CPMS mobile application, patient vital signs can be tracked remotely, offering doctors reliable access to critical health information at any time. The adaptability of the system to real-life conditions—such as sharing datasets for validation—makes it versatile for diverse healthcare environments. Additionally, different ML methods [27, 30] were systematically validated, with the ensemble approach delivering the most accurate predictions. This highlights the potential of hybrid models in predicting patient conditions effectively.

6. FUTURE SCOPE

Looking ahead, the system could be expanded by integrating embedded devices capable of directly capturing live readings from ICU instruments such as ventilators, infusion pumps, and cardiac monitors. This would enable more comprehensive patient monitoring and reduce manual data entry.

Further development may also focus on connecting additional patient monitoring systems through embedded technologies and real-time operating systems, thereby improving scalability and openness of healthcare infrastructure [2]. Another important area for advancement is predicting patient mortality risks using ML models, which is already under ongoing research.

Future work will emphasize enhancing both the accuracy and efficiency of models to align with evolving healthcare needs. Expanding scalability will also allow doctors to monitor multiple patients remotely, while families can receive timely updates without frequent hospital visits. These improvements will make healthcare services more accessible, reduce the burden on medical facilities, and enhance the overall patient experience.

REFERENCES

[1] Gardner R.M., Shabot M.M. (2006) Patient-Monitoring Systems. In: Shortliffe E.H., Cimino J.J. (eds) Biomedical Informatics. Health Informatics. Springer, New York, NY

[2] Aggarwal, M., & Madhukar, M. (2017). IBM's Watson Analytics for Health Care: A Miracle Made True. In Cloud Computing Systems and Applications in Healthcare (pp. 117-134). IGI Global.

[3] "Rational Unified Process", URL: [online] Available: https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf.

[4] Anwar Islam, Tuhin Biswas. Health System in Bangladesh: Challenges and Opportunities. American Journal of Health Research. Vol. 2, No. 6, 2014, pp. 366-374. doi: 10.11648/j.ajhr.20140206.18

[5] P. Griffiths, A. R. Saucedo, P. Schmidt, G. Smith. Vital signs monitoring in hospitals at night. (n.d.). Retrieved from https://www.nursingtimes.net/clinical-archive/assessment-skills/vitalsigns-monitoring-in-hospitals-at-night/5089989.article.

[6] An Embedded, GSM based, Multiparameter, Realtime Patient Monitoring System and Control – An Implementation for ICU Patients. Kumar, R., & Rajasekaran, M. P. (2016, January). An IoT based patient monitoring system using raspberry Pi. In 2016 International Conference on Computing Technologies and Intelligent Data Engineering (ICCTIDE'16) (pp. 1-4). IEEE.

[7] Nejkar, V. A., Nimbhorkar, S. R., Paliwal, J. K., & Shrivastav, A. A. (2018). Smart Nanny an IoT Based Baby Monitoring System. iManager's Journal on Computer Science, 6(1), 28.

[8] Ruiz, V. M., Saenz, L., Lopez-Magallon, A., Shields, A., Ogoe, H. A., Suresh, S., & Tsui, F. R. (2019). Early Prediction of Critical Events for Infants with Single Ventricle Physiology in Critical Care Using Routinely Collected Data. The Journal of Thoracic and Cardiovascular Surgery.

[9] Lin, K., Hu, Y., & Kong, G. (2019). Predicting In-hospital Mortality of Patients with Acute Kidney Injury in the ICU Using Random Forest Model. International Journal of Medical Informatics.

[10] Teres, D., Lemeshow, S., Avrunin, J. S., & Pastides, H. A. R. R. I. S. (1987). Validation of the mortality prediction model for ICU patients. Critical care medicine, 15(3), 208-213.

[11] Ahmed, S. (n.d.). BREAST CANCER: PRESENTATION AND LIMITATION OF TREATMENT – BANGLADESH PERSPECTIVE. doi:10.4172/1948-5956.S1.041

[12] Clarke, F & Mcdonald, Ellen & Griffith, Lauren & Cook, D & Mead, M & Guyatt, G & Rabbat, Christian & Geerts, W & Arnold, D & Warkentin, T & Crowther, Mark. (2004). Thrombocytopenia in medical–surgical ICU patients. Critical Care. 8. 1-1. 10.1186/cc2592.

[13] Choi, N. G., DiNitto, D. M., & Kim, J. (2014). Discrepancy Between Chronological Age and Felt Age: Age Group Difference in Objective and Subjective Health as Correlates. Journal of Aging and Health, 26(3), 458–473. https://doi.org/10.1177/0898264314523449.

[14] Db2 Warehouse. (n.d.). Retrieved from https://www.ibm.com/usen/marketplace/db2-warehouse.

[15] Das, Sudipto & Nishimura, Shoji & Agrawal, Divyakant & El Abbadi, Amr. (2010). Live Database Migration for Elasticity in a Multitenant Database for Cloud Platforms.

[16] Lewis, D. D. (1998, April). Naive (Bayes) at forty: The independence assumption in information retrieval. In European conference on machine learning (pp. 4-15). Springer, Berlin, Heidelberg.

[17] Dreiseitl, S., & Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. Journal of biomedical informatics, 35(5-6), 352-359.

[18] Du, W., & Zhan, Z. (2002, December). Building decision tree classifier on private data. In Proceedings of the IEEE international conference on Privacy, security and data mining-Volume 14 (pp. 1-8). Australian Computer Society, Inc.

[19] Ghate, V. N., & Dudul, S. V. (2010). Optimal MLP neural network classifier for fault detection of three phase induction motor. Expert Systems with Applications, 37(4), 3468-3481.

[20] Graczyk, M., Lasota, T., Trawiński, B., & Trawiński, K. (2010, March). Comparison of bagging, boosting and stacking ensembles applied to real estate appraisal. In Asian conference on intelligent information and database systems (pp. 340-350). Springer, Berlin, Heidelberg.

[21] What is the Jupyter Notebook? (n.d.). Retrieved from https://jupyternotebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html.

[22] Getting started with SMS Gateway. (n.d.). Retrieved from https://www.ibm.com/support/knowledgecenter/en/SS4U29/sms_getting started.html.

[23] Tucker, L. R., & Lewis, C. (1973). A reliability coefficient for maximum likelihood factor analysis. Psychometrika, 38(1), 1-10.

[24] Lu, Y., Cohen, I., Zhou, X. S., & Tian, Q. (2007, September). Feature selection using principal feature analysis. In Proceedings of the 15th ACM international conference on Multimedia (pp. 301-304). ACM.

[25] Rish, I. (2001, August). An empirical study of the naive Bayes classifier. In IJCAI 2001 workshop on empirical methods in artificial intelligence (Vol. 3, No. 22, pp. 41-46).

[26] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). Applied logistic regression (Vol. 398). John Wiley & Sons.

[27] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In Advances in neural information processing systems (pp. 2951-2959).

[28] Muja, M., & Lowe, D. G. (2009). Fast approximate nearest neighbors with automatic algorithm configuration. VISAPP (1), 2(331-340), 2.

[29] Naidoo, L., Cho, M. A., Mathieu, R., & Asner, G. (2012). Classification of savanna tree species, in the Greater Kruger National Park region, by integrating hyperspectral and LiDAR data in a Random Forest data mining environment. ISPRS journal of Photogrammetry and Remote Sensing, 69, 167-179.

[30] Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. In Advances in neural information processing systems (pp. 2962-2970).

[31] Mistry, J., & Inden, B. (2018). An approach to sign language translation using the Intel Realsense camera.

[32] Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine learning, 40(2), 139-157.

[33] Baumann, K. (2003). Cross-validation as the objective function for variable-selection techniques. TrAC Trends in Analytical Chemistry, 22(6), 395-406.

[34] Overview: Estimators, transformers and pipelines - spark.ml. (n.d.). Retrieved from https://spark.apache.org/docs/1.6.0/ml-guide.html.