Comprehensive Evaluation of Supervised Machine Learning Classifiers in Stroke Prediction

[Censored due to GitHub] `Work by NastyPigz`

Abstract

Stroke ranks as the 2nd leading cause of death according to the Global Burden of Diseases (GDB)[4]. With the rise of modern technology, machine learning became a popular approach in medical predictions[21]. However, imbalanced medical datasets present as an ongoing challenge, significantly decreasing the potential performance of machine learning models [9]. This study aims to investigate, analyze, and compare the performance of supervised machine learning classifiers, including logistic regression, linear discriminant analysis, k-nearest neighbours, decision tree classifier, stochastic gradient descent, and ensemble classifiers, which consists of random forest classifier, adaptive boosting algorithm, and voting classifier. According to the analysis results, logistic regression and stochastic gradient descent work best as base classifiers for the imbalanced stroke dataset, while the adaptive boosting algorithm work best amongst the ensemble algorithms.

Keywords: stroke prediction, supervised machine learning, machine learning classifiers, imbalanced dataset, medical prediction, hyperparameter tuning

1 Introduction

Cerebral stroke is clinically defined as "an acute, focal neurological deficit attributed to vascular injury of the central nervous system"[16], also known as cerebrovascular accident (CVA)[23]. According to the Global Burden of Diseases, Injuries, and Risk Factors (GBD)'s case study in 2019, stroke is the second leading cause of death and the third-leading cause of disability globally, accounting for approximately 11.6% of total deaths[4]. Despite modern medical advancements, 87% of strokes are sudden ischemic infarctions, striking rapidly without any warning signs[12] and costing lives before the provision of any medical assistance. The other type of stroke, hemorrhagic stroke, is less common and strikes slightly slower than ischemic but is more likely to be fatal. Therefore, early diagnosis is vital in preventing strokes, potentially increasing mortality rate.

Early diagnosis of medical conditions often rely on the analysis of risk factors. Stroke patients experience a variety of risk factors, which can be split into non-modifiable and modifiable risk factors. Non-modifiable risk factors include age, sex, ethnicity, and genetics, while modifiable risk factors include hypertension, diabetes, smoking, and other sedentary behaviours[16]. These information are utilized for both stroke treatment and diagnostic processes.

The topic of stroke prediction can date as far back as 1991, when the Framingham Heart Study developed the Stroke Risk Factor Prediction Chart to estimate the chance of the stroke based on factors such as age, sex, and smoking status[11]. However, limitations may apply as the chart would only work for patients aged 54-86 over a 10-year period. In 1996, one can find written records on stroke prediction by magnetic resonance imaging (MRI), such as the research done by Warach et al. on diffusion/perfusion MRI [25]. Although the outcomes are desirable, one might question the cost of such procedure. Hence, there requires a less costly approach.

Due to the influence of modern technology, machine learning introduced itself as a prominent approach in medical prediction[21], with methods ranging from linear regression to artificial neural networks. Machine learning (ML) is known as "an umbrella term that refers to a broad range of

algorithms that perform intelligent predictions based on a data set". In other words, computers calculating outcomes based on input[17]. Machine learning can effectively predict based on physiological data, classifying data into predefined labels, minimizing the cost compared to MRI procedures[22]. In the field of stroke prediction with machine learning, one can go as far back as 2010, where Khosla et al. published "An integrated machine learning approach to stroke prediction"[10]. In the study conducted by Khosla et al, Support Vector Machines (SVMs) were utilized, and used on the Cardiovascular Health Study (CHS) dataset. However, the traditional SVM approaches are outperformed by more recent classifiers such as K-Nearest Neighbours, Random Forest, and Decision Tree[5], potentially increasing time efficiency. Similar SVM approaches can be seen in recent studies as well, including "A comparative analysis of machine learning classifiers for stroke prediction: A predictive analytics approach" by Biswas et al. from late 2022[3].

Existing research conducted on stroke prediction mostly involves complete and class balance datasets, although medical datasets are usually incomplete and imbalance[14]. Despite the frequent nature of missing data, ML algorithms can handle them through procedures such as imputation, which denotes a procedure that replaces the missing values by some plausible values[1]. There are lots of existing studies on the imputation of missing values, some notable ones include Liu et al.'s hybrid approach to stroke prediction[14], which picks features and imputes using Random Forest, and a comprehensive analysis of different methods such as SMOTE, Focal Loss, and PCA-KMeans done by Jing[8]. Nevertheless, the primary focus of this study is the comparative analysis of different supervised machine learning classifiers. Therefore a simple imputation using mean or median is sufficient.

Machine learning classifiers demonstrate distinct behaviours depending on different configurations. Hence, it is crucial to tune the parameters into ensuring best model performance[20]. Cross-validation (CV) is a data resampling method for "estimating the true prediction error of models and to tune model parameters"[2]. CV is usually used to tackle a common machine learning problem called overfitting, which occurs when the accuracy becomes too high due to training on the same dataset, resulting in an ungeneralized model. To fix this, CV splits the training and testing data into k folds and uses them for training and testing in turns. This way, a better balance can be found on the classifier that works well for new, unseen data. Hyperparameters are parameters that are not directly learnt within estimators, and are passed in programmatically. Since there are lots of variance in performance while using different hyperparameters, one must effectively determine the best combination, often based on a scoring criteria. One such method is grid search, which performs an exhaustive search over a range of specified parameter values for an estimator, trying every possible combination and returning the best parameters based on a set scoring criteria[13].

When it comes to measuring the performance of the machine learning classifiers, different metrics are used. As outlined by Yacouby et al., the selection of a metric involves factors such as interpretability, computational cost, differentiability, and popularity in a specific field[26]. Notably, metrics such as accuracy, precision, and recall are easily interpretable, making them relevant in this study. By default, the python library used in this study, sklearn, uses "binary" as average for precision, recall, and f1. However, using the default parameter results in an accuracy of zero. Therefore, a weighted average will be used.

This research aims to explore and evaluate the performance of various supervised machine learning classifiers in stroke prediction employing an imbalanced stroke prediction dataset from Kaggle[24]. The classifiers will predict the likelihood of stroke occurrence based on parameters such as gender, age, body mass index (BMI), medical history, and smoking status. These classifiers consist of logistic regression, linear discriminant analysis, k-nearest neighbours classifier, decision tree classifier, stochastic gradient descent, random forest classifier, adaptive boosting with decision tree as base, and voting classifier.

2 Methodology

2.1 Dataset description

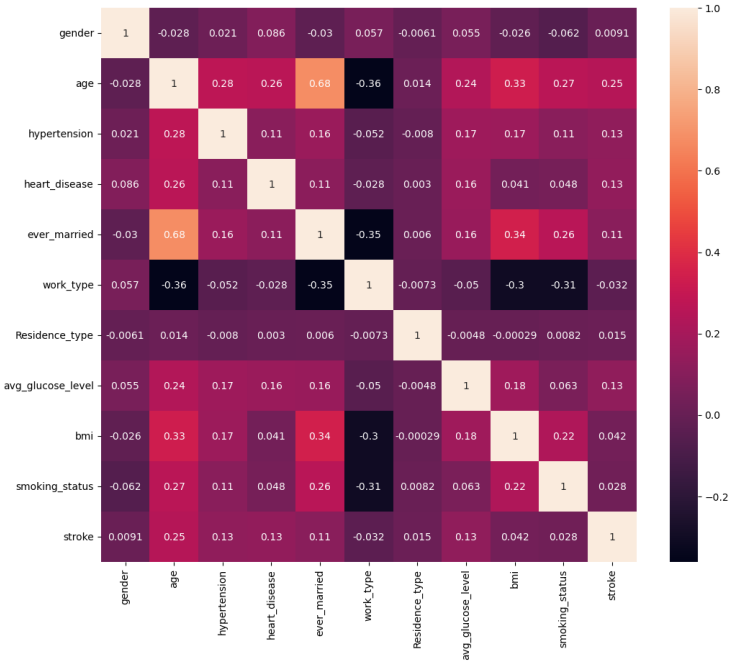
In this study, the dataset used is provided by Kaggle, uploaded by a user named *Fedesoriano*. The dataset used is detailed in Table 1, consisting of 10 input features, 1 unique identifier, and 1 target feature. The dataset consists of 5110 unique entries and no duplicate records. In regards

to missing values, the dataset is found to only contain 201 missing values in the bmi column, accounting for approximately only 4% of the entire dataset. In addition, the "smoking status" column includes an enum labelled "Unknown". Rather than dropping or imputing this enum, it will be employed as a categorical feature label.

Table 1: Dataset Details

FEATURE	DESCRIPTION
id	Patient unique identifier (number)
gender	Male/Female/Other (string enum)
age	Patient age (0.08 - 82)
hypertension	whether or not patient had hypertension (Yes/No)
heart_disease	whether or not patient had heart disease (0/1)
ever_married	whether or not patient is married (0/1)
work_type	Private/Self-employed/Govt_job/Never_worked/children (string enum)
residence_type	Urban/Rural (string enum)
avg_glucose_level	Average blood glucose level (55.1 - 272)
bmi	Body mass index (10.3 - 97.6)
smoking_status	Unknown/Never/Smokes/Former (string enum)
stroke	whether or not patient has stroke (0/1)

Figure 1: A heatmap of the correlation in the dataset used generated by seaborn.



Finally, it is worth noting that the training and testing data is split into 80% and 20% respectively, with random state 42.

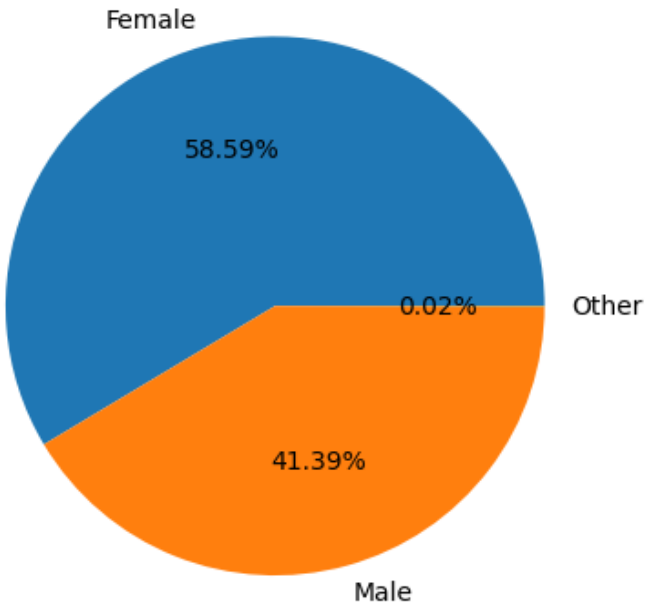
2.2 Preprocessing

The preprocessing procedure is divided into three parts - filtering, imputing, and encoding. The filtering process includes the removal of duplicate values, removal of outliers, and the removal

of other redundant values. Since there are no duplicate values in this dataset, this step can be skipped and continue with the removal of outliers. In machine learning, a well-known and simple approach for outlier removal is visually spotting them. Because this study primarily focuses on the comparative analysis of classifiers, the visualization process will be used.

In Figure 2, it can be shown that there is only 1 record with gender ”other” out of the total 5110, accounting for only 0.02% of the data.

Figure 2: Pie chart of gender distribution in dataset



In the study conducted by Jing, it can be shown that age and bmi also have their respective outlier ranges. In Jing’s study, age less than 25 and bmi larger than the 60% percentile range is removed. However, it is important for the models to successfully predict stroke in the 0 to 25 age range, as well as accounting for high bmi. Therefore, these data will not be removed as outliers. Finally, the filtering process ends with the removal of redundant values. It can be observed that the unique identifier ”id” serves no purpose, and is therefore redundant and removed.

After filtering, the process of imputing begins. Since this study primarily focuses comparing different supervised machine learning classifiers, imputation will be done with a simple imputer using the median strategy. The selection of the median strategy is completely random, as mean and median values are very close to each other in the dataset used. Past studies have used imputation methods such as SMOTE, Focal Loss + DNN, PCA-Kmeans (Jing), maximization algorithms based Bayesian classification (Liu et al.), and etc. Table of numerical values is demonstrated below in Table 2

Table 2: Traits of meaningful numerical values

FEATURE	MEAN	STD
age	43.23	22.61
avg_glucose_level	106.14	45.29
bmi	28.89	7.85

After imputation, the data are encoded using label encoder to fit each column with a categorical label. The results of label encoder is demonstrated in Table 3.

Table 3: Results after label encoding

FEATURE	LABELS	COUNT
gender.female	0	2994
gender.male	1	2115
ever_married.Yes	1	3353
ever_married.No	0	1756
work_type.Private	2	2925
work_type.Self-employed	3	819
work_type.children	4	687
work_type.Govt_job	0	657
work_type.Never_worked	1	22
Residence_type.Urban	1	2596
Residence_type.Rural	0	2514
Smoking."never smoked"	2	1892
Smoking.Unknown	0	1544
Smoking."formerly smoked"	1	885
Smoking.smokes	3	789

After encoding the data, there is one step left before passing data into the classifier models, that being feature scaling. To improve the performance of classifiers, standard scaler is used in a pipeline before Logistic Regression (LR) and Linear Discriminant Analysis (LDA). Standard scaler is configured with the default parameters, with both mean and standard deviation. This step is instrumental in achieving feature scaling, ensuring that each feature adheres to a standardized distribution with zero mean and unit variance. Notably, it is worth mentioning that K-Nearest Neighbours is a classifier known to work better with standard scaling. However, the classifier demonstrates lower accuracy when used with standard scaler, and is therefore not used. Finally, it is worth noting that the mathematical formula of standardization is $z = \frac{x-\mu}{\sigma}$, with mean defined as $\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i)$, and standard deviation defined as $\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2}$ [19].

2.3 Classifiers

The classifiers utilized in this study includes Logistic Regression (LR), Linear Discriminant Analysis (LDA), K-Nearest Neighbours (KNN), Decision Tree Classifier (DTC), Random Forest Classifier (RFC), Stochastic Gradient Descent (SGD) Classifier, AdaBoost Classifier, and Voting Classifier. Classifying these models in accordance with the classification structure in the Scikit-learn library, the distribution contains 2 linear models, 1 tree model, 1 discriminant analysis model, 1 neighbor model, and 3 ensemble models. This selection creates a well-rounded combination of methods, each designed for specific data traits, aiding in drawing out the best predictive understandings from the dataset[18].

2.4 Hyperparameter tuning

To achieve optimal predictive performance from the classifiers, a systemic approach using hyperparameter tuning is used. In this study, the hyperparameter tuning process is done with the popular cross-validation approach using the GridSearchCV function provided by the sklearn python library. Hyperparameters are configurable settings that influence the behaviour of the machine learning classifier models. Grid search involves exploring a predefined range of values for each hyperparameter, resulting in a list of parameter combinations. At the same time, cross-validation is used to assess the model's performance under the settings. In cross validation, the dataset is partitioned into subsets, where each alternates as a validation set while the remaining is used for training. Additionally, accuracy is used for scoring to evaluate model performance. Finally, the hyperparameter used for this study can be found in Table 4. (Default is the default in the python library sklearn)

Table 4: Range of parameters used in GridSearchCV

PARAMETER	OPTIONS
lr.penalty	l1, l2
lr.C	1e-4 1e-3 1e-2 1e-1 1 1e1 1e2 1e3 1e4
lr.solver	lbfgs, liblinear, sag, saga, newton-cg, newton-cholesky
lr.multi_class	ovr, multinomial
lr.class_weight	None, balanced
lr.tol	1e-4, 1e-3, 1e-2
lda.solver	svd, lsqr, eigen
lda.shrinkage	auto, None, 0.1, 0.5, 0.9
lda.tol	1e-4, 1e-3, 1e-2
knn.algorithm	auto, ball_tree, kd_tree, brute
knn.leaf_size	list(range(20, 40))
knn.n_neighbors	list(range(1, 25))
knn.p	list(range(10))
knn.weights	uniform, distance
dtc.max_depth	None, 5, 10, 20
dtc.min_samples_split	2, 5, 10
dtc.min_samples_leaf	1, 2, 5
dtc.max_features	None, sqrt, log2, 0.2, 0.5
dtc.max_leaf_nodes	None, 5, 10, 20
dtc.criterion	gini, entropy, log_loss
dtc.min_impurity_decrease	0.0, 0.01, 0.05, 0.1

Table 5: Range of parameters used in GridSearchCV (continued)

PARAMETER	OPTIONS
rfc.n_estimators [1]	100, 200, 300
rfc.bootstrap	True, False
sgd.loss	hinge, log_loss, modified_huber, squared_hinge, perceptron, huber, squared_error, epsilon_insensitive, squared_epsilon_insensitive
sgd.penalty	l1, l2
sgd.learning_rate	constant, optimal, invscaling, adaptive
sgd.alpha	0.0001, 0.001, 0.01, 0.1
sgd.class_weight	None, balanced
sgd.tol	1e-3, 1e-4, 1e-5
sgd.eta0	1e-1, 1e-2, 1e-3, 1e-4, 1e-5
AdaBoost + Optimized DTC .n_estimators	50, 100, 150, 200, 250, 300, 350, 400, 450, 500
AdaBoost + Optimized DTC .learning_rate	1, 2, 3, 4, 5

The voting classifier was not listed in the table, since it is searched iteratively rather than using grid search and cross validation. Both soft and hard voting methods are tested, and a list of combinations consisting of three previously optimized classifiers is input into the estimator parameter. Note that SGD is excluded from the list of combinations during soft voting, since the optimized parameter "epsilon_insensitive" does not support probability estimates. Furthermore, AdaBoost is excluded from voting as it is already an ensemble classifier.

Finally, after all the hyperparameters are tuned, each classifier is evaluated based on their accuracy score, precision score with weighted average, recall score with weighted average, and f1 score with weighted average.

3 Results and Discussion

3.1 Results

Below lists the best hyperparameters after tuning with grid search cross-validation (default is default value in sklearn)

1. LR(solver="liblinear", C=0.0001, penalty="l1")
2. LDA(default)
3. KNN(n_neighbors=6, p=1)
4. DTC(criterion="entropy", max_depth=5, max_features=0.2)
5. RFC(criterion="entropy", max_depth=5, max_features=0.2)
6. SGD(loss="epsilon_insensitive", alpha=0.01)
7. AdaBoostClassifier(estimator=DTC(4), learning_rate=4, n_estimators=300)
8. VotingClassifier(estimators=[KNN(3), LDA(2), RFC(5)], method="soft")

The random states of LR, RFC, GCD, and the DTC inside AdaBoost are all 42. The random state for DTC is 73787, and the random state for AdaBoost is 45147. [2]

In Table 6, an outline is shown that elaborates on the performance achieved by each classifier.

[2] *Note that LDA, KNN, and Voting does not use random state.*

Table 6: Performance results of each classifier

CLASSIFIER	ACCURACY	PRECISION	RECALL SCORE	F1 SCORE
LR(...)	96.48%	93.08%	0.9648	0.9475
LDA(default)	95.90%	93.94%	0.9578	0.9476
KNN(...)	96.36%	93.08%	0.9638	0.9470
DTC(...)	96.28%	93.07%	0.9628	0.9465
RFC(...)	96.48%	93.08%	0.9648	0.9475
SGD(...)	96.48%	93.08%	0.9648	0.9475
AdaBoost(DTC, ...)	96.67%	96.78%	0.9667	0.9521
Voting(..., method="soft")	96.58%	96.69%	0.9658	0.9499

3.2 Discussion

Among the base classifiers, LR and SGD have the highest accuracy score, while LDA has the lowest accuracy score. However, LDA has the highest F1 and precision score, indicating that it's better at correctly identifying true positives. In the ensemble classifiers, AdaBoost achieved the highest accuracy, followed by the voting classifier, and the random forest classifier. What's interesting about the boosting and voting classifiers is that their precision scores are significantly higher than the base classifiers, making them much more reliable at identifying true positives. In addition, the voting classifier beats all base classifiers, and exhibits interesting behaviour as LDA, the least accurate classifier, is among the estimators used in the voting classifier to achieve maximum accuracy. The other two estimators are K-Nearest Neighbours and Random Forest Classifier, making the list diverse as each uses very different approaches. Lastly, the Random Forest Classifier exhibits interesting behaviour as it has the same accuracy score as the top base classifiers. Although, due to resource and time limitations, RFC could only be tuned with two ranging parameters bootstrap and n_estimators, with the rest filled by the tuned hyperparameters used in Decision Tree Classifier.

In comparison to past research, this study aligns with observations made by Jing, wherein classifiers such as LR, SGD, AdaBoost, and KNN are among the most accurate classifiers. Although, the accuracy score in this study is significantly higher. This may be due to the different approaches in preprocessing, as Jing employed several imputing algorithms, filter out more rows of data, and used One Hot encoding instead of labelling. Overall, both studies seem to demonstrate generalization. When compared to the study done by Biswas et al., similar patterns emerge, particularly when classifiers RFC, KNN, LR, AdaBoost, and Voting Classifier all demonstrate similar rankings in accuracy. Notably, the outcomes in Biswas et al.'s study seems to be less generalized, as obvious overfitting can be seen with an accuracy of 99.99% on the SVM model.

4 Conclusion

To summarize, this research compares and analyzes the performance of numerous supervised machine learning classifiers. Through an examination of diverse classifiers, it has become evident that each method offers distinct strengths and limitations, as demonstrated in the voting classifiers and results. These findings may enable healthcare professionals to make more informed decisions in identifying individuals at risk of stroke, all while minimizing the cost. It also demonstrates the potential of machine learning to improve healthcare systems. However, it is essential to acknowledge the evolving nature of healthcare data and its implications on model performance. Future research may focus on the exploration of hybrid models, fine-tuned for specific groups such as demographics, as well as addressing the challenges posed by incomplete and imbalanced data.

References

[1] Batista, G. E. A. P. A., & Monard, M. C. (2003). An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5-6), 519-533. https://doi.org/ 10.1080/713827181

[2] Berrar, D. (2019). Cross-Validation. https://www.researchgate.net/profile/Daniel-Berrar/publication/324701535_Cross-Validation/links/5cb4209c92851c8d22ec4349/Cross-Validation.pdf

[3] Biswas, N., Uddin, K. M. M., Rikta, S. T., & Dey, S. K. (2022). A comparative analysis of machine learning classifiers for stroke prediction: A predictive analytics approach. *Healthcare Analytics, 2*, 100116. https://doi.org/10.1016/j.health.2022.100116

[4] Feigin, V. L., Stark, B. A., Johnson, C. O., Roth, G. A., Bisignano, C., Abady, G. G., Abbasifard, M., Abbasi-kangevari, M., Abd-allah, F., Abedi, V., Abualhasan, A., Abu-rmeileh, N. M., Abushouk, A. I., Adebayo, O. M., Agarwal, G., Agasthi, P., Ahinkorah, B. O., Ahmad, S., Ahmadi, S., . . . Almustanyir, S. (2021). Global, regional, and national burden of stroke and its risk factors, 1990–2019: A systematic analysis for the global burden of disease study 2019. *The Lancet Neurology, 20*(10), 795-820. https://doi.org/10.1016/s1474-4422(21)00252-0

[5] García, S., Luengo, J., & Herrera, F. (2015). *Data preprocessing in data mining.* Springer International. https://doi.org/10.1007/978-3-319-10247-4

[6] Hawkins, D. M. (1980). *Identification of outliers*. Springer Netherlands. https://doi.org/10.1007/ 978-94-015-3994-4

[7] James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An introduction to statistical learning: With applications in Python.* Springer.

[8] Jing, Y. (2022, November 14). Machine Learning Performance Analysis to Predict Stroke Based on Imbalanced Medical Dataset. Retrieved August 12, 2023, from https://doi.org/10.48550/ arXiv.2211.07652

[9] Kaur, H., Pannu, H. S., & Malhi, A. K. (2019). A systematic review on imbalanced data challenges in machine learning. *ACM Computing Surveys, 52*(4), 1-36. https://doi.org/10.1145/3343440

[10] Khosla, A., Cao, Y., Lin, C. C.-Y., Chiu, H.-K., Hu, J., & Lee, H. (n.d.). An integrated machine learning approach to stroke prediction. *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 183-192. https://doi.org/10.1145/ 1835804.1835830

[11] Koton, S., Schneider, A. L. C., Rosamond, W. D., Shahar, E., Sang, Y., Gottesman, R. F., & Coresh, J. (2014, July 16). Stroke incidence and mortality trends in US communities, 1987 to 2011. *JAMA, 312*(3), 259. https://doi.org/10.1001/jama.2014.7692

[12] Kuriakose, D., & Xiao, Z. (2020). Pathophysiology and treatment of stroke: Present status and future perspectives. *International Journal of Molecular Sciences, 21*(20), 7609. https://doi.org/ 10.3390/ijms21207609

[13] Liashchynskyi, P., & Liashchynskyi, P. (2019, December 12). Grid search, random search, genetic algorithm: A big comparison for NAS. Retrieved August 12, 2023, from https://doi.org/10.48550/ arXiv.1912.06059

[14] Liu, T., Fan, W., & Wu, C. (2019). A hybrid machine learning approach to cerebral stroke prediction based on imbalanced medical dataset. *Artificial Intelligence in Medicine, 101*, 101723. https://doi.org/10.1016/j.artmed.2019.101723

[15] Malik, S., Harous, S., & El-sayed, H. (2020). Comparative analysis of machine learning algorithms for early prediction of diabetes mellitus in women. *Modelling and Implementation of Complex Systems*, 95-106. https://doi.org/10.1007/978-3-030-58861-8_7

[16] Murphy, S. J., & Werring, D. J. (2020). Stroke: Causes and clinical features. *Medicine, 48*(9), 561-566. https://doi.org/10.1016/j.mpmed.2020.06.002

[17] Nichols, J. A., Herbert Chan, H. W., & Baker, M. A. B. (2019). Machine learning: applications of artificial intelligence to imaging and diagnosis. *Biophysical Reviews, 11*(1), 111-118. https://doi.org/10.1007/s12551-018-0449-9

[18] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., & Dubourg, V. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research, 12*, 2825-2830.

[19] Raschka, S. (2014, July 11). *About feature scaling and normalization.* Sebastian Raschka, PhD. Retrieved August 12, 2023, from https://sebastianraschka.com/Articles/ 2014_about_feature_scaling.html

[20] Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., & Brenning, A. (2019). Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. *Ecological Modelling, 406,* 109-120. https://doi.org/10.1016/j.ecolmodel.2019.06.002

[21] Sidey-gibbons, J. A. M., & Sidey-gibbons, C. J. (2019). Machine learning in medicine: A practical introduction. *BMC Medical Research Methodology, 19*(1). https://doi.org/10.1186/ s12874-019-0681-4

[22] Sirsat, M. S., Fermé, E., & Câmara, J. (2020). Machine learning for brain stroke: A review. *Journal of Stroke and Cerebrovascular Diseases, 29*(10), 105162. https://doi.org/10.1016/ j.jstrokecerebrovasdis.2020.105162

[23] *Stroke, cerebrovascular accident.* (n.d.). World Health Organization Eastern Mediterranean Regional Office. Retrieved August 11, 2023, from https://www.emro.who.int/health-topics/ stroke-cerebrovascular-accident/index.html

[24] *Stroke prediction dataset.* (2021, January 26). Kaggle. Retrieved August 3, 2023, from https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset

[25] Warach, S., Dashe, J. F., & Edelman, R. R. (1996). Clinical outcome in ischemic stroke predicted by early diffusion-weighted and perfusion magnetic resonance imaging: A preliminary analysis. *Journal of Cerebral Blood Flow & Metabolism, 16*(1), 53-59. https://doi.org/10.1097/ 00004647-199601000-00006

[26] Yacouby, R., & Axman, D. (2020). Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems* (pp. 79-91). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.eval4nlp-1.9