Machine Learning for Stroke Prediction: Model Development and Evaluation

Ioannis Stathakis
School of Science and Engineering
University of Nicosia
Athens, Greece

jstatdata@gmail.com

ABSTRACT

This project explores the use of machine learning to predict stroke occurrence based on patient health records. After performing extensive data cleaning and exploratory analysis, we trained and evaluated a variety of classification models using four different sampling strategies to address the target class imbalance: Full SMOTE, Reduced SMOTE, Downsampling and Class-Weighting. The modeling phase included classic classifiers (e.g., Logistic Regression (Binary), Support Vector Machines (SVMs), Random Forest), neural networks (FFNNs, Perceptron), and a transfer-learning approach (TabNet). Each model was assessed using metrics such as recall, ROC-AUC, and specificity, with particular emphasis on recall due to the high cost of false negatives in a healthcare context such as this one. After comparing performance across strategies, the final model selected was the SVM with linear kernel and the class-weighting sampling strategy, offering the best combination of high recall and strong overall generalization. A flow-chart of the project's structure and procedures can be seen in *Figure 1*.

KEYWORDS

Stroke Prediction, Machine Learning, Classification, Oversampling, Class Imbalance, Logistic Regression, Random Forest, Support Vector Machines, Transfer Learning

1. Introduction

Stroke is one of the leading causes of death and long-term disability worldwide, making early detection and prevention a critical public health goal **(1)**. In recent years, machine learning has shown promise in assisting medical decision-making by identifying patterns that may not be obvious to human clinicians. This project focuses on predicting stroke occurrence based on a set of clinical and demographic features, using real-world patient data. The aim was to explore various modeling approaches, evaluate their effectiveness under different data sampling strategies, and ultimately select the most suitable classifier for this binary prediction task. In doing so, the project also considers the importance of class imbalance, feature relevance, and the balance between sensitivity (recall), specificity and generalization.

2. Dataset Overview and Preparation

The dataset contained a mix of continuous, categorical and binary features. Several standard preprocessing steps were applied, such as dealing with various formatting issues and redundant features, properly imputing missing values and handling outliers. The cleaned dataset was saved and served as the foundation of the exploratory and modeling phases.

2.1. Dataset Summary

The dataset **(2)** consists of 5,110 patient records, each describing demographic, lifestyle, and medical information related to stroke risk. It includes twelve original features and one binary target variable, Stroke. Predictor variables include both numerical/continuous features (e.g., Age, Avg_Glucose_Level, BMI) and categorical features (e.g., Gender, Work_Type, Smoking_Status). An engineered feature, Age_Group, was added to the features set, based on the continuous Age variable, to assist in grouped statistical analysis and missing values imputation. Stroke occurrence is relatively rare in the dataset, appearing in approximately 5% of cases.

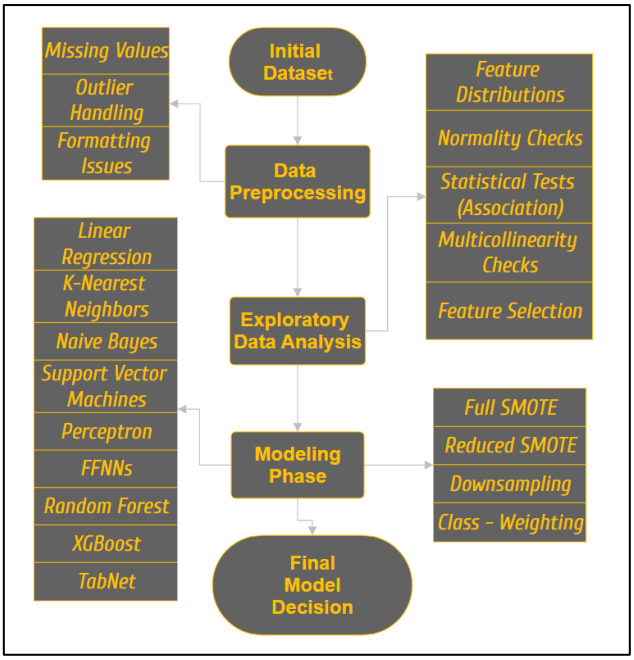

Figure 1: *Flow chart depicting the various processes and steps of the project.*

2.2. Dataset Preprocessing

Redundant columns and very low-frequency missing values, which did not have an impact on the target, were removed. To address the missing values of the BMI column specifically, group-wise imputation was performed with the utilization of the median BMI value, within each age group. This approach helped preserve meaningful differences across age groups, which could be relevant to stroke risk. Boxplots and Interquartile Range (IQR) Analysis were used to detect outliers in continuous features. Outliers in the BMI feature were retained due to their medical relevance and minimal impact on skewness. The Average Glucose Level feature contained more extreme outliers and exhibited a bimodal distribution *(Figure 2)*. This bimodality is further discussed in Section 3. The decision was made to also retain this natural bimodal distribution (and the outliers contained in the feature) to preserve those potentially medically important patterns.

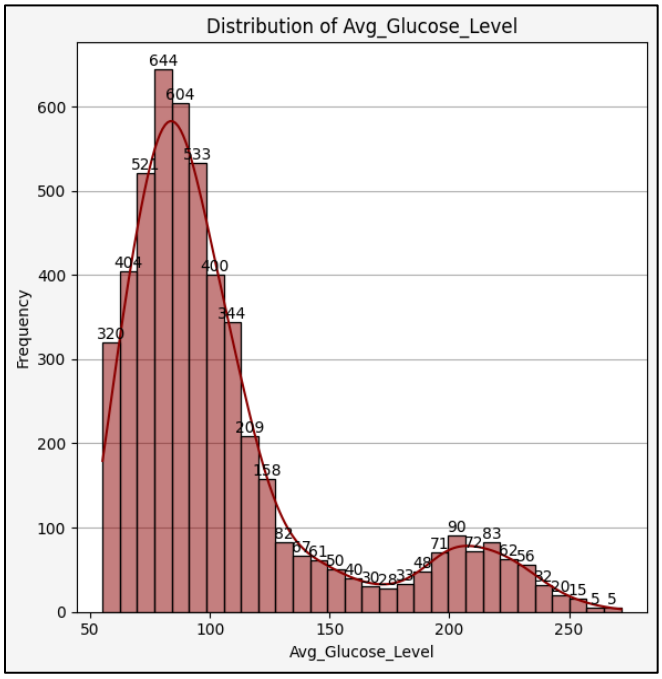

Figure 2: *Average Glucose Level distribution histogram, where the bimodality of the feature is quite evident, even before the confirmation via more robust statistical testing.*

3. Exploratory Data Analysis (EDA)

The goal of this phase is to explore data and classes/subclasses distributions, class imbalance, correlations, and relationships between features and also between potential predictors and the binary target variable.

3.1. Normality Tests and Feature Distributions

For continuous features, normality was examined using the Shapiro-Wilk and Anderson-Darling tests **(3, 4)**, as well as visual QQ-Plots. Examination revealed that variables such as Age, Avg_Glucose_Level, and BMI did not follow a normal distribution. The bimodal distribution displayed by the Average Glucose Level feature, in particular, led to its temporary categorization into two groups using Gaussian Mixture Models (GMMs) for further analysis **(5)**. The two groups created by the GMMs were used to further assess the predictive value and importance of this variable towards the target but ultimately were not maintained as separate features in the dataset. Kernel Density Estimation (KDE) plots and Boxplots were used to compare the distributions of numerical variables across stroke classes. Notably, patients who had a stroke were found to be generally older *(Figure 3)* and exhibited higher average glucose levels.

3.2. Statistical Testing

Categorical features were tested for association with the Stroke variable using the Chi-Squared Test of Independence **(6)**. Features such as Hypertension, Heart_Disease, and Work_Type showed statistically significant differences in Stroke proportions. The non-parametric Mann-Whitney U test **(7)** was used to compare the distributions of non-normal features between stroke and non-stroke groups. The test confirmed statistically significant differences across these variables, further supporting their inclusion in the modeling process. Correlation heatmaps (Pearson and Spearman) were employed to explore potential multicollinearity **(8)** among continuous/non categorical variables such as Age, Avg_Glucose_Level, and BMI, which showed no problematic linear correlations.

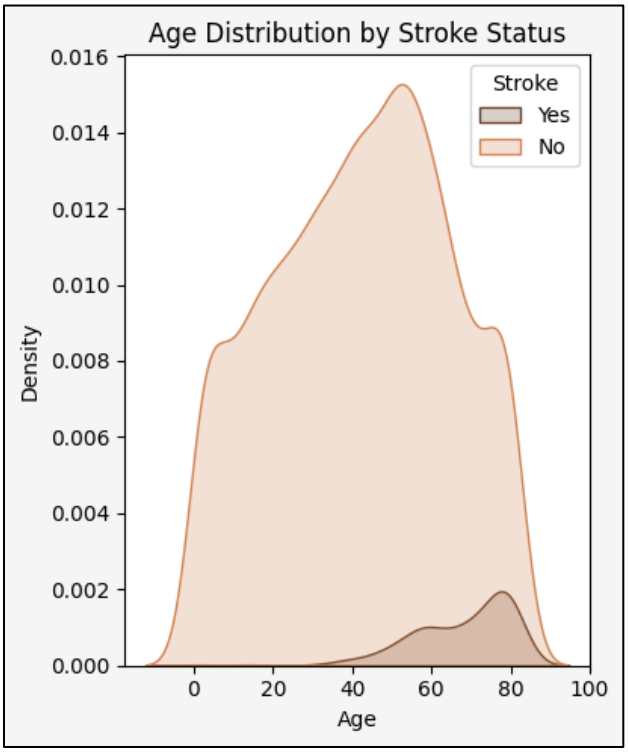

Figure 3: *Kernel Density Estimation (KDE) plot depicting the difference of the Age feature distribution between the Stroke target variable subclasses.*

3.3. Feature Selection

Mutual Information (MI) analysis **(9)** was conducted to evaluate the dependency between each feature and the target variable in a non-linear and non-parametric way. This analysis reinforced the importance of features such as Age, Hypertension, Heart_Disease, and glucose-related measures. Feature relevance was further validated through a Random Forest Classifier **(10)**, which provided importance scores (Tree-based Learning). The results consistently highlighted Age, Avg_Glucose_Level, and BMI as the most informative continuous features, along with binary indicators such as Hypertension, Heart_Disease, and specific categories of Work_Type and Smoking_Status. These insights shaped the final feature selection strategy to be used during the next model development phase.

4. Modeling and Evaluation Methodology

The modeling phase involved the development and evaluation of machine learning classifiers to predict Stroke occurrence using the information from the features we have analyzed up to this point. The modeling workflow emphasized proper data partitioning, class imbalance handling, consistent and appropriate performance metrics, and comparative model evaluation. A selection of various models was used, with the aim of identifying the most effective approach, balancing accuracy, sensitivity, and interpretability.

4.1. Modeling Validation and Sampling Strategies

The considerably high imbalance of the target variable has a significant effect on classification performance. The way this issue is addressed is by focusing on specific metrics (such as sensitivity (recall), specificity, AUC) instead of accuracy, which in such cases can actually be misleading **(11)**. Various sampling strategies during the modeling phase are also employed, such as SMOTE-based oversampling **(12)**, downsampling **(13)** and model-internal class-weighting schemes **(14)**, to ensure the models are not biased towards the majority class.

The following sampling strategies were used, specifically:

- *Full SMOTE*: Oversampling the minority class to achieve a 50-50 class balance for the target variable.
- *Reduced SMOTE*: Oversampling to a 40:60 ratio, to avoid excessive use of synthetic data.
- *Downsampling*: Reducing the majority target class to match the minority more closely.
- *Class-Weighting*: automatic model class-weighting during training (abbreviated as "Weighted" on the results graphs).

All models were trained using an 80/20 train/test split. In some cases, potential overfitting was assessed, using either train vs test recall bar plot comparisons, or test vs validation recall monitoring over training epochs (in the neural network trials), or training loss vs validation AUC over training epochs (in the TabNet transfer learning trials).

4.2. Models Employed

The models which were used for experimentation can be grouped into five characteristic categories:

1. *Baseline Models*: Simple, interpretable models used as performance benchmarks. There were Logistic Regression, Naïve Bayes and K-Nearest Neighbors (K-NN) models.
2. *Support Vector Machines (SVMs)*: Margin-based classifiers with strong performance on tabular data. In these trials we used two types of SVMs, Linear and RBF Kernel.
3. *Tree-Based Models*: Models based on decision trees structures. There was a Random Forest and an XGBoost **(15)** model used in these trials.
4. *Neural Network Models*: Models inspired by brain architecture, used for non-linear patterns. Here we employed a Two-Class Averaged Perceptron, and a Feedforward Neural Network (FFNN).
5. *Deep Learning/Transfer Learning for Tabular Data*: Advanced models designed for structured data with attention-based learning. The model used in these trials was TabNet **(16)**.

4.3. Modeling Stages

The first stage involved using the Full SMOTE sampling strategy, and the Logistic Regression, Naïve Bayes, K-NN, SVM, Random Forest and XGBoost models. Metrics were assessed after each training session, with particular focus on recall, specificity, and AUC, as already mentioned. After this initial stage, the decision was made to abandon models which seemed to be struggling with overfitting, and which were less potent performers. The models which were selected from the above list as more promising were the Logistic Regression, SVM (Linear Kernel) and SVM (RBF Kernel), and these were the ones used with the rest of the sampling strategies.

The second, third and fourth stages involved using the Reduced SMOTE, Downsampling and Class-Weighted sampling strategies respectively, with the three stronger performing models mentioned in the previous paragraph. Overfitting was also assessed for all three models and sampling strategies in these stages, using Train vs Test Recall barplots and comparisons. Although overfitting was observed in the Full SMOTE and Reduced SMOTE stages, it seemed to no longer be an issue in the Downsampling and Class-Weighted stages. This could be explained by the models learning the SMOTE synthetic data well but then struggling with the differences and the "noise" of the real-world data.

Stage five involved an assessment of the best performing models up to that point, and employing hyperparameter tuning for further experimentation, using only the best model – sampling strategy pairs. These were noted to be the following:

1. Logistic Regression + Downsampling
2. SVM (Linear Kernel) + Class-Weighting
3. SVM (RBF Kernel) + Reduced SMOTE

In these hyperparameter tuning trials, the SVM (Linear Kernel) had the highest recall, good specificity, close to top AUC and stable performance (across all stages). The Logistic Regression model was respectable but still behind the SVM (Linear Kernel). Finally, the SVM (RBF Kernel) showed extreme overfitting and collapsed recall metric.

Finally, stages six and seven involved the trials conducted with the neural network models (Perceptron and FFNN) and the TabNet deep learning model, as mentioned in paragraph 4.2. All these models were trained using the three of our four sampling strategies, namely Full SMOTE, Reduced SMOTE and Downsampling. However, none of these models and sampling strategy combinations managed to outperform the simpler models of the previous stages. Results for the best performing models overall will be discussed in more detail, in the next section.

5. Results/Discussion

This project explored the application of machine learning to predict stroke occurrence using clinical and demographic data. Throughout the process, various models were evaluated across different sampling strategies to handle class imbalance, with a particular focus on improving sensitivity (recall), specificity and AUC. In addition to traditional/baseline models, Neural Networks and a transfer learning approach with TabNet were also explored. While some of these

approaches achieved high recall in specific setups, they often required more tuning and delivered less stable results.

More specifically, in the Full SMOTE trial the Perceptron classifier achieved solid and stable results, with a recall of 0.62 and specificity at 0.81. The standard FFNN initially trained over 20 epochs performed slightly below the Perceptron, achieving a recall of 0.54. An interesting observation, however, was that the FFNN seemed to begin overfitting after the 4th epoch. This led to the decision to continue the trials with a 4-epoch training limit. Training the FFNN with only 4 epochs led to a significant boost in recall (0.90), indicating that shorter training cycles helped prevent overfitting and allowed the model to better capture the signal from the minority class.

This boost in recall, however, came with accompanying substantial drops in accuracy, specificity and precision, and these results were also very similar to the results of the Neural Network run using the Reduced SMOTE strategy. This might be understandable, given the high imbalanced nature of the dataset. Since SMOTE oversampling was applied in training only (and not on the test-set), models might be overly "eager" to predict positives, even when applied to an imbalanced real-world test distribution.

The final Neural Network trial using the Downsampling strategy ended with both models showing a considerable drop in recall, and with the overfitting check revealing that the FFNN had completely erratic behavior (no clear overfitting pattern in validation recall) in consecutive epochs during training *(Figure 4)*. This might indicate high variance in model learning from batch to batch, which is common with smaller datasets like downsampled sets.

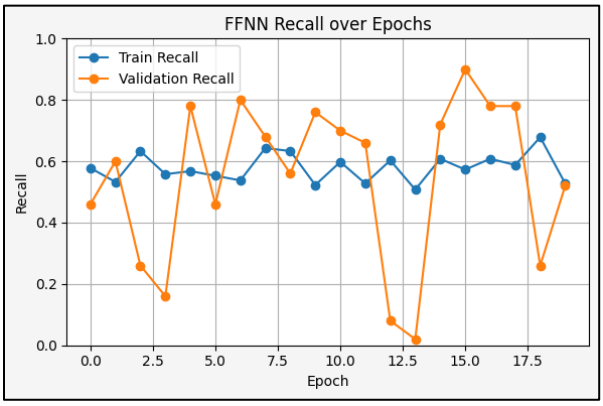

Figure 4: *The erratic inconsistency of the FFNN validation recall over the 20 training periods (under the Downsampling strategy).*

Throughout all four sampling strategies and combinations, the models which showed the best, most consistent, and balanced results were the SVMs with Linear and RBF kernels *(metric examples in Figures 5 and 6)*. The SVM model with Linear Kernel, trained using the Class-Weighted sampling strategy, proved to be the most balanced and reliable performer *(Figures 7 and 8)*. This model achieved the highest recall (0.88) of all tested classifiers, while maintaining a strong AUC (0.85) and specificity (0.68). Additional important aspects which cannot be overlooked are the fact that it did not require synthetic data or extensive hyperparameter tuning, and the fact that it showed no signs of overfitting *(Figure 9)*, making it a robust and interpretable choice. Given the project's priority on minimizing false negatives in stroke prediction, this model was selected as the final choice for deployment. A final summary of the best performing model per sampling strategy can be found in *Figure 10.*

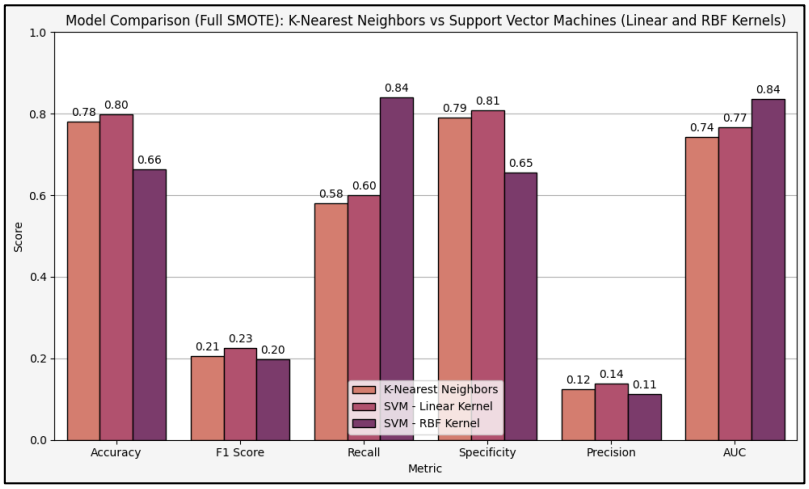

Figure 5: *Metrics bar plot performance comparison of the K-NN and SVM models, under the Full SMOTE sampling strategy.*

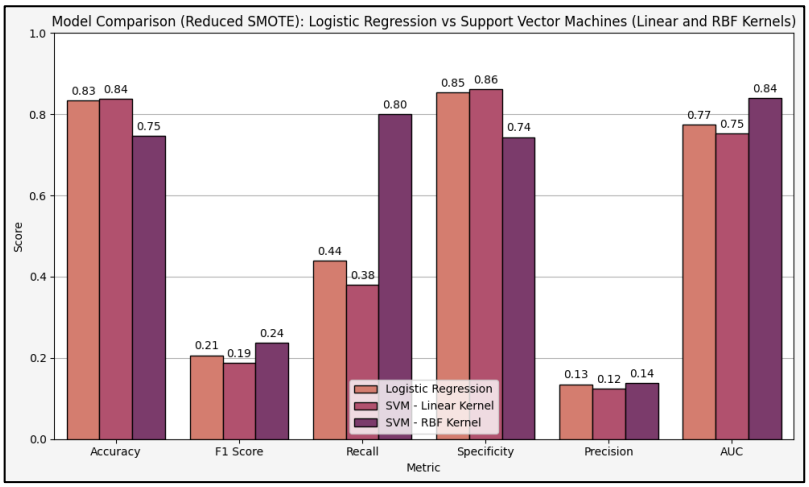

Figure 6: *Metrics bar plot performance comparison of the Logistic Regression and SVM models, under the Reduced SMOTE sampling strategy.*

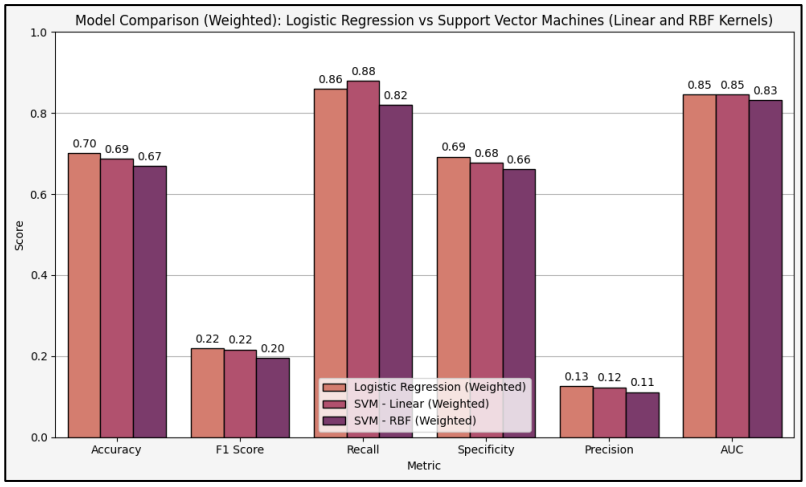

Figure 7: *Metrics bar plot performance comparison of the Logistic Regression and SVM models, under the Class-Weighted sampling strategy.*

Overall, the project demonstrated that simpler models, when paired with well-chosen preprocessing strategies, can often outperform more complex alternatives in structured medical datasets. Future work could involve deploying the model of choice in a real-world clinical setting and also incorporating additional patient historical features.

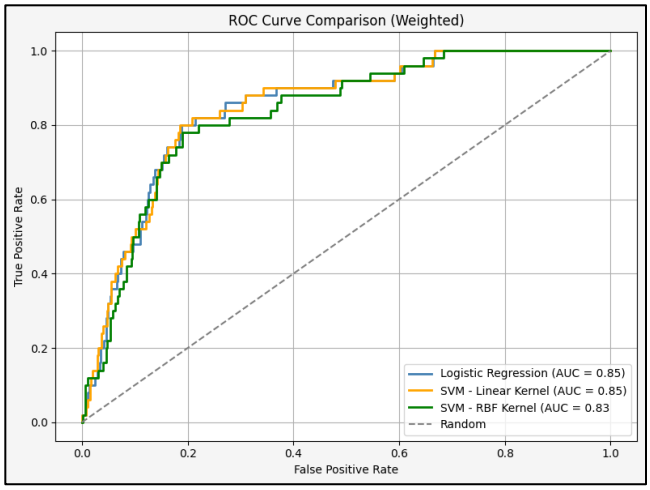

Figure 8: *ROC-AUC comparison of the Logistic Regression and SVM models, under the Class-Weighted sampling strategy.*

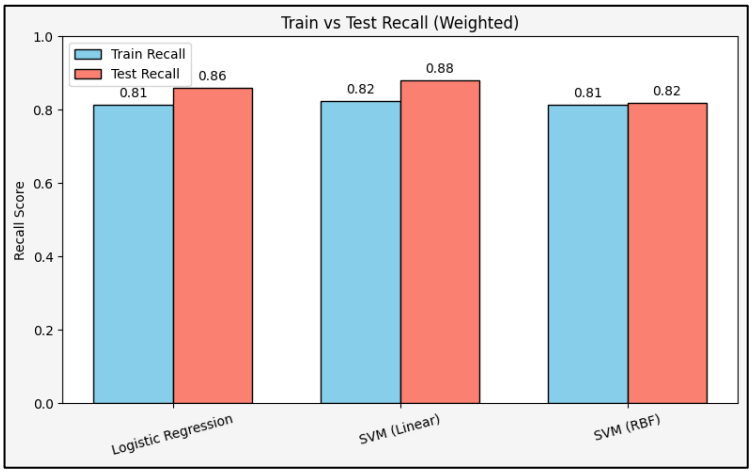

Figure 9: *Overfitting check (Train vs Test recall comparison) for the Logistic Regression and SVM models, under the Class-Weighted sampling strategy.*

Sampling Strategy	Best Model	Recall	Specificity	AUC
Full SMOTE	SVM (RBF)	0.82	0.662	0.833
Reduced SMOTE	SVM (RBF)	0.8	0.744	0.84
Downsampling	SVM (RBF)	0.82	0.719	0.84
Weighted	SVM (Linear - Weighted)	0.88	0.677	0.846

Figure 10: *Best performing model per sampling strategy, considering recall, specificity and AUC.*

References

1. Stroke Overview | National Institute of Neurological Disorders and Stroke [Internet]. [cited Apr 21, 2025]. Available from: https://www.ninds.nih.gov/health-information/stroke/stroke-overview.

2. Stroke Prediction Dataset [Internet]. []. Available from: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset.

3. Anderson–Darling Test for Normality [Internet].; 2025 [updated -04-20; cited Apr 22, 2025]. Available from: https://en.wikipedia.org/w/index.php?title=Anderson%E2%80%93Darling_test&oldid=1286589624.

4. Shapiro–Wilk Test for Normality [Internet].; 2025 [updated -04-20; cited Apr 22, 2025]. Available from: https://en.wikipedia.org/w/index.php?title=Shapiro%E2%80%93Wilk_test&oldid=1286590067.

5. Scikit-Learn: Gaussian Mixture Models [Internet]. [cited Apr 21, 2025]. Available from: https://scikit-learn/stable/modules/mixture.html.

6. Chi-Squared Test Of Independence (Feature Selection) [Internet]. [cited Apr 21, 2025]. Available from: https://scikit-learn/stable/modules/generated/sklearn.feature_selection.chi2.html.

7. Mann–Whitney U Test [Internet].; 2025 [updated -04-08; cited Apr 21, 2025]. Available from: https://en.wikipedia.org/w/index.php?title=Mann%E2%80%93Whitney_U_test&oldid=1284635664.

8. Multicollinearity [Internet].; 2025 [updated -04-09; cited Apr 21, 2025]. Available from: https://en.wikipedia.org/w/index.php?title=Multicollinearity&oldid=1284764086.

9. Scikit-Learn: Mutual Information Classifier [Internet]. [cited Apr 21, 2025]. Available from: https://scikit-learn/stable/modules/generated/sklearn.feature_selection.mutual_info_classif.html.

10. Random Forest Classifier (Feature Importance) [Internet]. [cited Apr 21, 2025]. Available from: https://scikit-learn/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

11. Why Accuracy Is Not A Good Metric For Imbalanced Data | Towards AI [Internet].; 2022 [updated August 11; cited May 2, 2025]. Available from: https://towardsai.net/p/l/why-accuracy-is-not-a-good-metric-for-imbalanced-data.

12. Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, W. Philip Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research. 2002 01 June;Vol 16(1):321–57.

13. What is downsampling? | IBM [Internet].; 2024 [updated -06-15; cited Apr 21, 2025]. Available from: https://www.ibm.com/think/topics/downsampling.

14. How Does the class_weight Parameter in Scikit-Learn Work? [Internet].; 2024 [updated -08-07; cited Apr 21, 2025]. Available from: https://www.geeksforgeeks.org/how-does-the-classweight-parameter-in-scikit-learn-work/.

15. Tiangi Chen CG. XGBoost: A Scalable Tree Boosting System. KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016 13 August.

16. Arik SO, Pfister T. TabNet: Attentive Interpretable Tabular Learning. 2020 December 9.