Stroke Risk Prediction

Daniel Jolin[†], Nathan Waters[‡], Andrew Nicola[§]
Department of Electrical and Computer Engineering
University of North Carolina at Charlotte
Charlotte, NC, US
[†]djolin1@charlotte.edu (801282735),
[‡]nwaters3@charlotte.edu (801283595), [§]anicola2@charlotte.edu (801136465)

Abstract—**Strokes are a leading cause of death, underscoring the critical need for early detection and prevention. This project explores the application of machine learning models to predict stroke risk using the Stroke Prediction Dataset from Kaggle. By leveraging Logistic Regression, Support Vector Machines (SVM), and Neural Networks, the study addresses key challenges such as data imbalance and feature scaling to optimize predictive accuracy. Results highlight trade-offs between model interpretability and performance, with Neural Networks achieving the best balance of precision and recall. This work demonstrates the potential for data-driven approaches to support proactive stroke prevention.**

I. Introduction

Strokes are among the leading causes of death, claiming approximately 6.5 million lives annually, which accounts for 11% of all deaths. The sudden and unpredictable nature of strokes underscores the urgency for rapid medical intervention, as an untreated stroke can result in the loss of 1.9 million neurons every minute.

This project aims to address this critical healthcare challenge by utilizing machine learning models to develop an early warning system. The goal is to create models that predict the likelihood of a stroke using parameters such as hypertension, age, history of heart disease, and other relevant factors, enabling timely and data-driven preventive measures.

By analyzing the ”Stroke Prediction Dataset” from Kaggle, the team explored various machine learning approaches, including Logistic Regression, Support Vector Machines (SVM), and Neural Networks. The project emphasizes data preprocessing properly, feature selection, and model optimization to deliver accurate and actionable predictions.

II. Dataset and Data Preprocessing

A. Dataset Description

The project utilized the *Stroke Prediction Dataset* sourced from Kaggle, containing patient-level data relevant to stroke prediction. The dataset consists of the following:

- **Samples**: The dataset includes 5,110 records, each representing an individual.
- **Features**: There are 12 primary features that capture demographic, medical history, and lifestyle attributes. These include:
  – Age, Gender, BMI
  – Hypertension, Heart Disease
  – Average Glucose Level
  – Residence Type, Smoking Status
  – Work Type (categorized into Private, Self-employed, Government Job, etc.)
  – Marital Status (Ever Married)

The target variable is a binary label indicating the occurrence of a stroke (`stroke: 1`) or the absence of it (`stroke: 0`).

TABLE I
Distribution of Stroke Cases in the Dataset

Category	Percentage	Description
1	4.9%	Cases with a stroke occurrence
0	95.1%	Cases without a stroke occurrence

B. Data Preprocessing

Proper data preprocessing was critical to ensure model performance and reliability. The following steps were taken to ensure accurate and effective model predictions:

1) **Handling Missing Values**:
   - The BMI feature contained missing values (NaN), which were replaced using mean imputation to avoid losing valuable records.
2) **Categorical to Numerical Conversion**:
   - Features like gender, job type, and smoking status were transformed into numerical formats using one-hot encoding or binary mapping to make them compatible with machine learning algorithms.
3) **Removing Non-Contributory Data**:
   - Records with `gender` labeled as ”Other” were excluded due to their limited representation.
4) **Balancing the Target Variable**:
   - The dataset was highly imbalanced, with stroke cases being an underrepresented group compared to non-stroke cases as seen in Table 1. To address this imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied to increase the representation of the minority class (stroke) in the training dataset, ensuring better learning for the models.

5) **Feature Scaling**:
   - All numerical features were scaled using `StandardScaler` to normalize their ranges, ensuring uniformity for models sensitive to scale, such as SVM and Neural Networks.

C. Feature Selection

A subset of relevant features was selected based on their predictive potential:

- **Demographics**: Age, Gender, Residence Type
- **Medical History**: Hypertension, Heart Disease, Average Glucose Level, BMI
- **Lifestyle Attributes**: Smoking Status, Work Type

The selected features formed the basis of the input for all machine learning models used in this project.

D. Data Splitting

The dataset was divided into training and testing subsets using an 80/20 split providing ample data for training while reserving a portion for unbiased evaluation. Balancing and preprocess as mentioned above was only applied to the training set to improve model learning, while the test set remained untouched to accurately asses the models' performance.

With the dataset prepared, the next step involved selecting and implementing machine learning algorithms for stroke risk prediction.

III. Modeling Approach

A. Overview of Algorithms

To predict stroke risk, three machine learning algorithms were implemented, each chosen for their unique strengths:

- **Logistic Regression**: Selected as a baseline model for its simplicity, allowing clear insights into the relationship between features and predictions.
- **Support Vector Machines (SVM)**: Used to explore both linear and non-linear decision boundaries, particularly effective with kernel methods.
- **Neural Networks**: Used an altered Res-Net architecture then later switched into a fully connected neural network to enhance performance.

B. Implementation Details

Each model was configured with tailored hyperparameters and optimization techniques to maximize performance:

- **Logistic Regression**:
  - Solver: `lbfgs`
  - Maximum Iterations: 1000
  - Class Weight: Balanced to address target imbalance
- **Support Vector Machines (SVM)**:
  - Kernel: Radial Basis Function (RBF)
  - Regularization Parameter (C): 1.0
  - Class Weight: Balanced
- **Res-Net Neural Network**:
  - Architecture: Input layer with 512 units, followed by three Residual Blocks, and a final output layer.
  - Optimizer: AdamW with a learning rate of 0.0002
  - Scheduler: OneCycleLR for dynamic learning rate adjustment
  - Early Stopping: Implemented to prevent overfitting
  - Loss Function: Focal Loss to mitigate the impact of class imbalance
  - Drop-Out: 0.20 after the input layer and each hidden layer
- **Fully Connected Neural Network**:
  - Architecture: Input layer, 3 hidden layers, sigmoid output.
  - Optimizer: AdamW, learning rate 0.0002.
  - Scheduler: OneCycleLR, max_lr 0.001.
  - Early Stopping: Stops after 15 epochs no improvement.
  - Loss Function: Focal Loss, handles class imbalance.
  - Drop-Out: 0.50 after the input layer and each hidden layer

C. Evaluation Metrics

Given the imbalanced nature of the dataset, performance was evaluated using the following metrics:

- **Accuracy**: The percentage of correct predictions made by the model.
- **Precision**: The percentage of predicted positives that are actually correct.
- **Recall**: The percentage of actual positives correctly identified by the model.
- **F1 Score**: The harmonic mean of precision and recall, used to balance their trade-off.

D. Challenges and Trade-offs

Several challenges and trade-offs emerged during model development:

- **Class Imbalance**: Stroke cases were underrepresented in the dataset, requiring techniques like SMOTE and class weighting to ensure models effectively learned from minority cases.
- **Precision vs. Recall**: Logistic Regression achieved high accuracy but struggled to identify stroke cases, while Neural Networks balanced precision and recall at the cost of slightly lower accuracy.
- **Complexity vs. Interpretability**: Neural Networks provided better performance but at the expense of interpretability compared to Logistic Regression.

With these models and strategies in place, the project proceeded to evaluate their performance and determine the best approach for stroke risk prediction.

IV. Results

The results of the models are summarized below, with metrics calculated for accuracy, precision, recall, and F1 score. For each algorithm, classification reports and confusion matrices are provided to offer detailed insights into the performance.

A. Logistic Regression

Logistic Regression demonstrated high overall accuracy but struggled with detecting stroke cases due to the class imbalance.

TABLE II
CLASSIFICATION REPORT FOR LOGISTIC REGRESSION

Class	Precision	Recall	F1-Score	Support
0	0.98	0.74	0.84	960
1	0.17	0.81	0.27	62
Accuracy	-	-	0.74	1022
Macro Avg	0.57	0.77	0.56	-
Weighted Avg	0.93	0.74	0.81	-

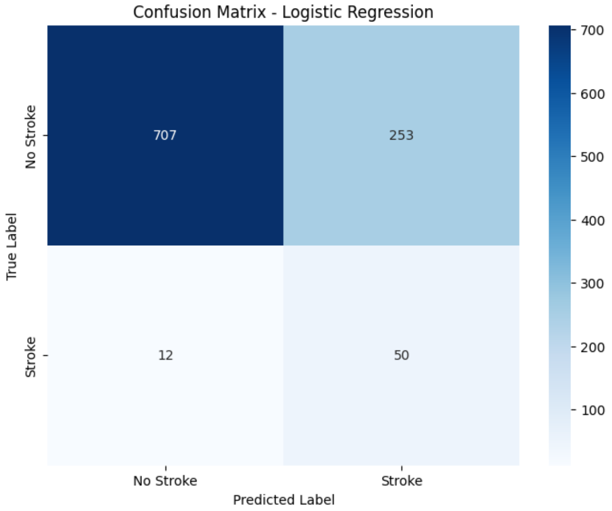

Fig. 1. Confusion Matrix for Logistic Regression

B. Support Vector Machines (SVM)

SVM provided a moderate balance between accuracy and recall, particularly effective in identifying non-stroke cases.

TABLE III
CLASSIFICATION REPORT FOR SVM

Class	Precision	Recall	F1-Score	Support
0	0.96	0.83	0.89	960
1	0.16	0.48	0.24	62
Accuracy	-	-	0.81	1022
Macro Avg	0.56	0.66	0.56	-
Weighted Avg	0.91	0.81	0.85	-

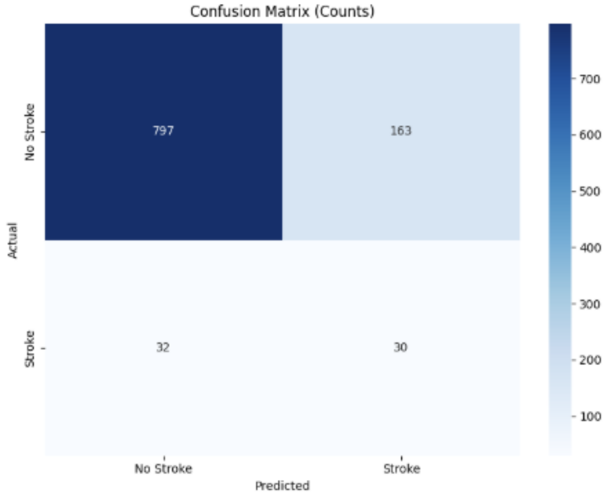

Fig. 2. Confusion Matrix for SVM

C. Res-Net Neural Network

The Res-Net Neural Network model achieved a balanced performance across metrics, showing improved recall for stroke cases at the expense of reduced accuracy. This model was trained over 100 Epoch. While this model performed better than SVM and Logistic Regression, it was lower than expected.

TABLE IV
CLASSIFICATION REPORT FOR RES-NET NEURAL NETWORK

Class	Precision	Recall	F1-Score	Support
0	0.95	0.91	0.93	960
1	0.14	0.24	0.18	62
Accuracy	-	-	0.86	1022
Macro Avg	0.55	0.57	0.55	-
Weighted Avg	0.90	0.86	0.88	-

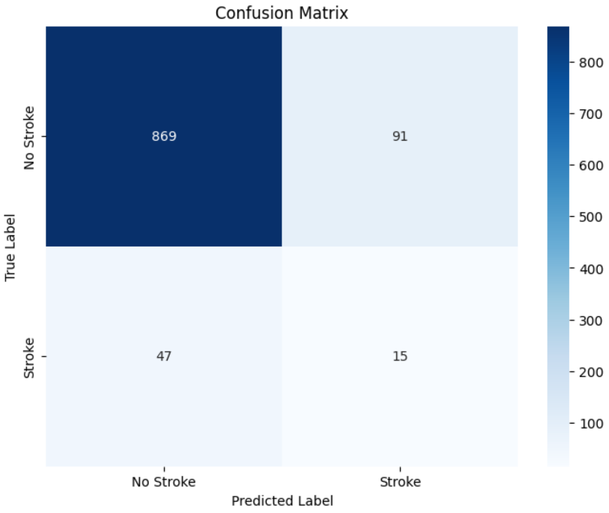

Fig. 3. Confusion Matrix for Res-Net Neural Network

D. Fully Connected Neural Network

The Fully Connected Neural Network stood out as the best model when compared to the results above. This model was trained over 100 Epoch.

TABLE V
CLASSIFICATION REPORT FOR FULLY CONNECTED NEURAL NETWORK

Class	Precision	Recall	F1-Score	Support
0	0.95	0.84	0.89	960
1	0.12	0.34	0.18	62
Accuracy	-	-	0.81	1022
Macro Avg	0.54	0.59	0.54	-
Weighted Avg	0.90	0.81	0.85	-

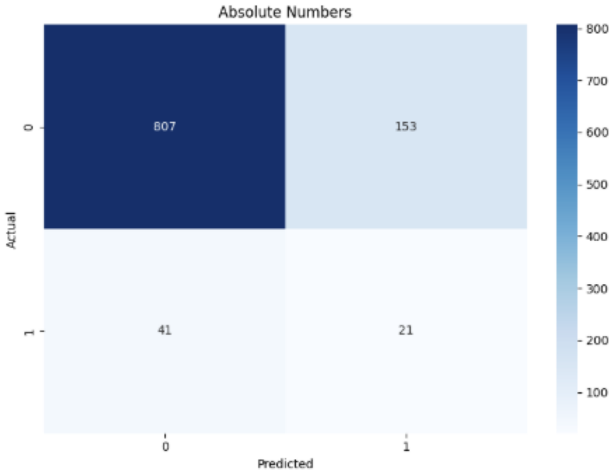

Fig. 4. Confusion Matrix for Fully Connected Neural Network

As mentioned above the Fully Connected Neural Network was trained over 100 Epoch, below in Fig 5, the performance of the model of each Epoch is graphed to visualize the models performance as it is trained.

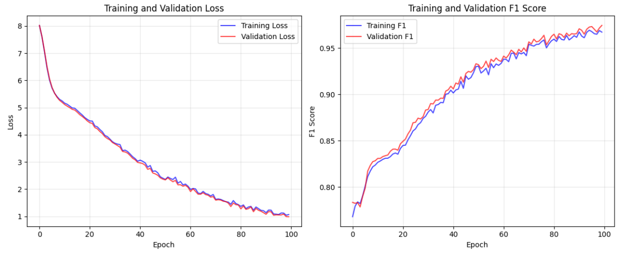

Fig. 5. Training and Validation Loss & F1 Score over Epoch

The Fully Connected Neural Network was coded to graph the feature importance at the end of the 100 Epoch as seen in Fig 6.

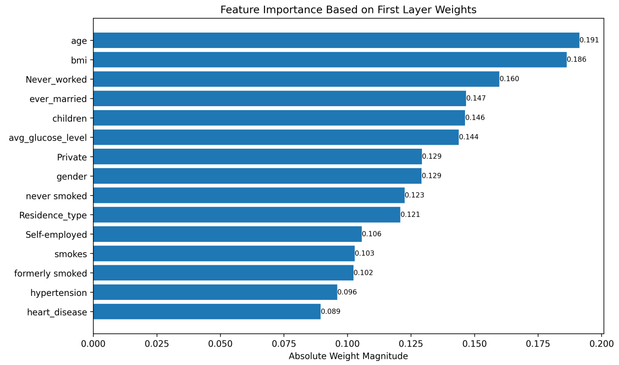

Fig. 6. Feature Importance

Overall, the results highlight the trade-offs between accuracy, precision, and recall across the four models. Logistic Regression served as a standard baseline for comparison, SVM showed robustness in distinguishing non-stroke cases, and the Neural Networks provided a better balance in detecting stroke cases.

V. LESSONS LEARNED AND FUTURE WORK

A. Summary of Findings

This project explored the use of machine learning algorithms to predict stroke risk based on patient data. The key findings include:

- Logistic Regression provided a standard baseline to compare to other models.
- Support Vector Machines (SVM) offered a moderate balance, excelling in distinguishing non-stroke cases but showing limitations in sensitivity.
- Neural Networks achieved the best balance between precision and recall, making it more effective in detecting stroke cases.

These results highlight the trade-offs inherent in predictive modeling for imbalanced medical datasets, where both sensitivity and specificity are critical.

B. Implications

The findings of this study have practical implications for stroke prevention. A machine learning-based early warning system could help healthcare providers identify high-risk individuals and implement preventive interventions, potentially saving lives. However, the effectiveness of such systems in real-world settings depends on addressing limitations in data quality and class imbalance.

C. Lessons learned

While the results are promising, several limitations must be acknowledged:

- The dataset was relatively small and imbalanced, which likely affected the models' generalizability.
- Important clinical features, such as family history and genetic factors, were not included in the dataset.
- The models were tested on a single dataset, and their performance on external data remains unevaluated.

D. Future Work

To build on this study, the following areas of future work are proposed:

- Collecting larger, more diverse datasets to improve model robustness and generalizability.
- Exploring advanced algorithms, such as ensemble methods or deep learning architectures, to enhance prediction performance.
- Incorporating additional features, such as genetic and environmental factors, for a more comprehensive risk assessment.
- Validating the models on external datasets and in real-world clinical trials to assess their practical applicability.

By addressing these limitations and expanding the scope of the research, future studies could significantly improve the accuracy and utility of machine learning models for stroke prediction.

VI. GitHub & Dataset

- Stroke Prediction Dataset on Kaggle
- GitHub Repository