STROKE PREDICTION USING MACHINE LEARNING LAB REPORT

Course: CSE422
Section: 17
Team Members:

Zawad Ahsan (23201136)
Abdullah Sajid Nafi (23101228)

Date: January 2026

Table of Contents

1. Introduction

This project predicts stroke risk in patients using supervised machine learning models. We trained and compared multiple algorithms using a publicly available healthcare dataset and evaluated them using accuracy, ROC-AUC, precision, recall, F1-score, and confusion matrices.

Project Objective

To implement and compare multiple machine learning models for predicting stroke risk based on health and demographic features, and identify the model best suited for early risk screening.

Methodology Overview

- Exploratory Data Analysis (EDA): Understand distributions, outliers, and feature relationships
- Data Preprocessing: Handle missing values, encode categorical variables, scale features
- Model Training: Train six ML algorithms with class imbalance handling
- Model Evaluation: Compare using accuracy, ROC-AUC, precision, recall, F1-score, and confusion matrices
- Model Selection: Choose the best model based on clinically meaningful metrics

2. Dataset Description

2.1 Dataset Overview

Dataset: Healthcare Stroke Data
Total Samples: 5,110 patient records
Total Features: 12 columns (11 input features + 1 target)
Task Type: Binary classification (stroke: 0 = No, 1 = Yes)

Features:

- Demographic: gender, age, ever_married, work_type, Residence_type
- Health Indicators: hypertension, heart_disease, avg_glucose_level, bmi
- Lifestyle: smoking_status
- Target Variable: stroke
- Identifier: id (not used for modeling)

Feature Type Summary:

- 7 Numerical features: id, age, hypertension, heart_disease, avg_glucose_level, bmi, stroke
- 5 Categorical features: gender, ever_married, work_type, Residence_type, smoking_status

2.2 Correlation Analysis

We computed correlations among numerical variables using Pearson, Spearman, and Kendall methods.

Key Findings:

- No strong correlations between features and stroke target
- Most correlations are weak but positive for clinically relevant variables

- Age shows the strongest association with stroke risk
- Hypertension and heart disease show moderate positive correlations

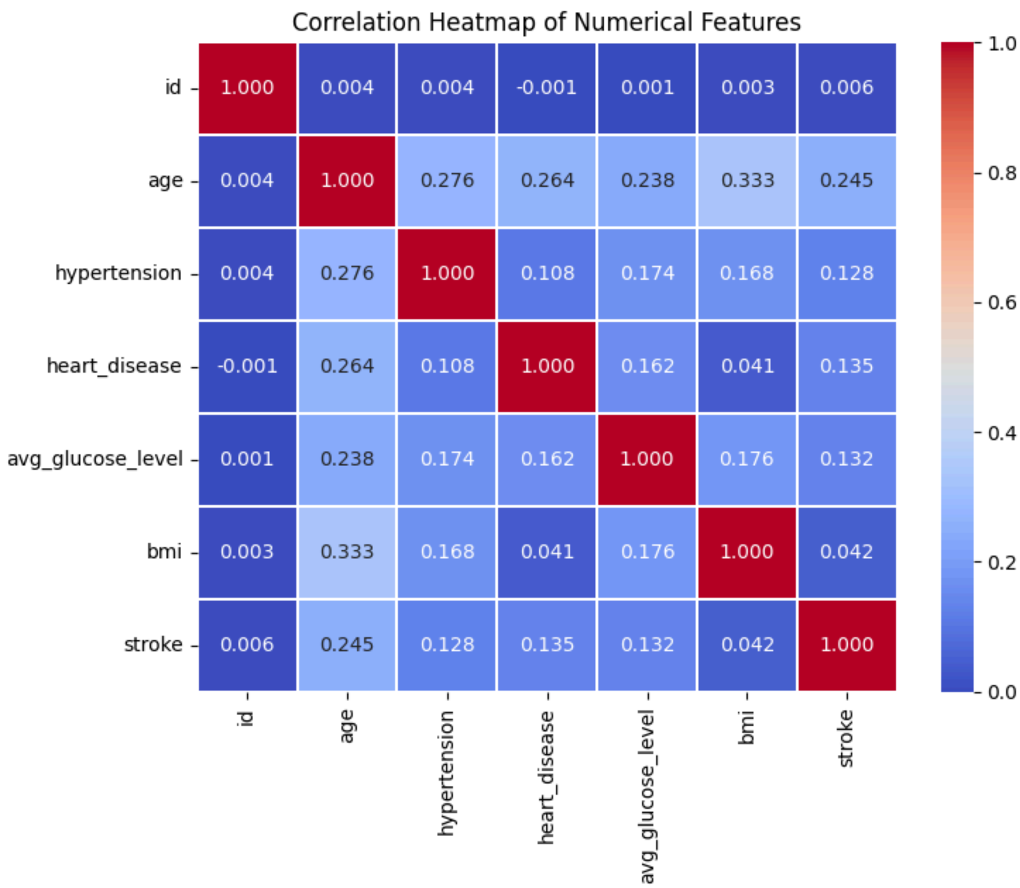

Figure 1:Pearson correlation heatmap

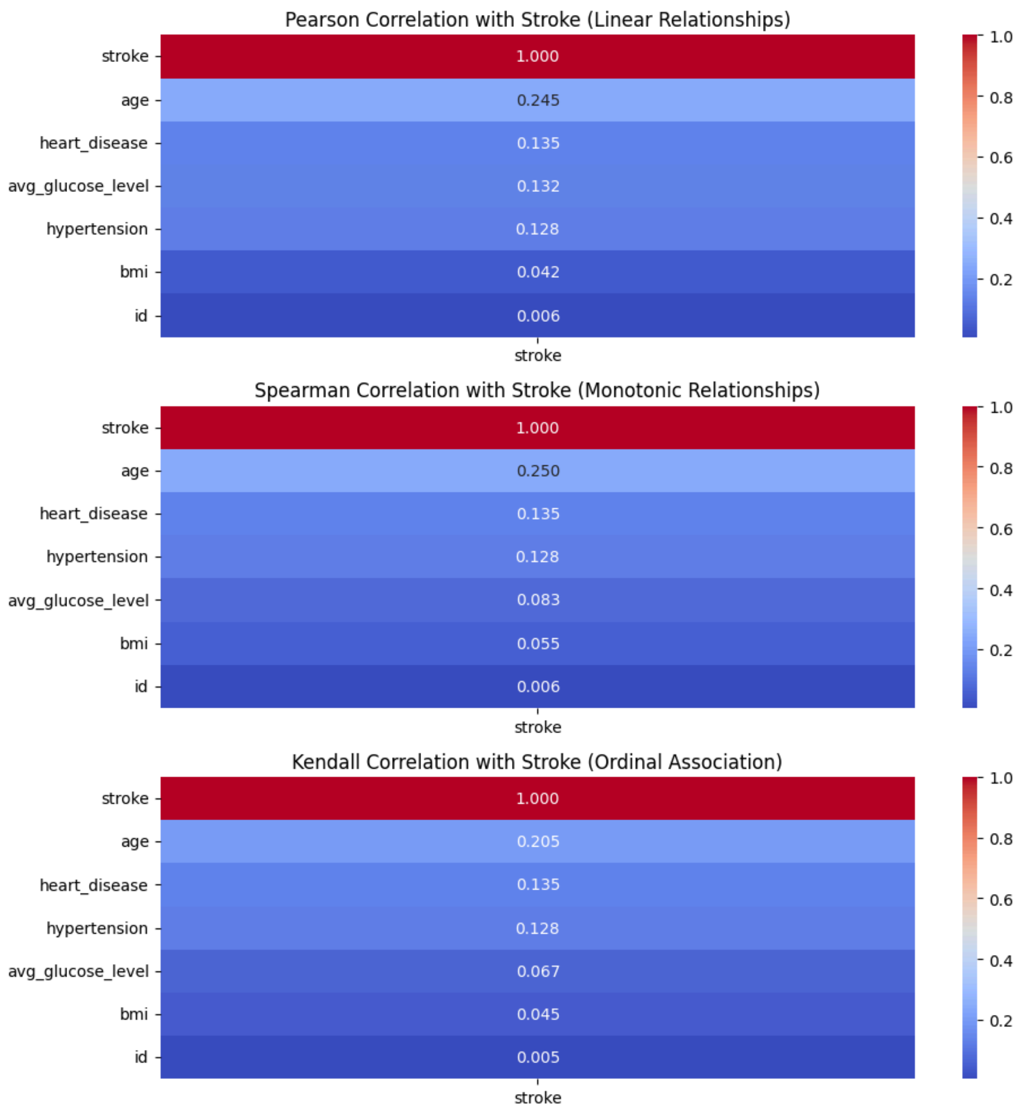

Figure 2: Pearson, Spearman, and Kendall correlation comparison

2.3 Class Imbalance Analysis

The target variable exhibits severe class imbalance:

- Class 0 (No Stroke): 4,861 samples (95.13%)
- Class 1 (Stroke): 249 samples (4.87%)

This 95:5 imbalance makes accuracy misleading as a sole metric—a model predicting all negatives achieves ~95% accuracy while detecting zero stroke cases.

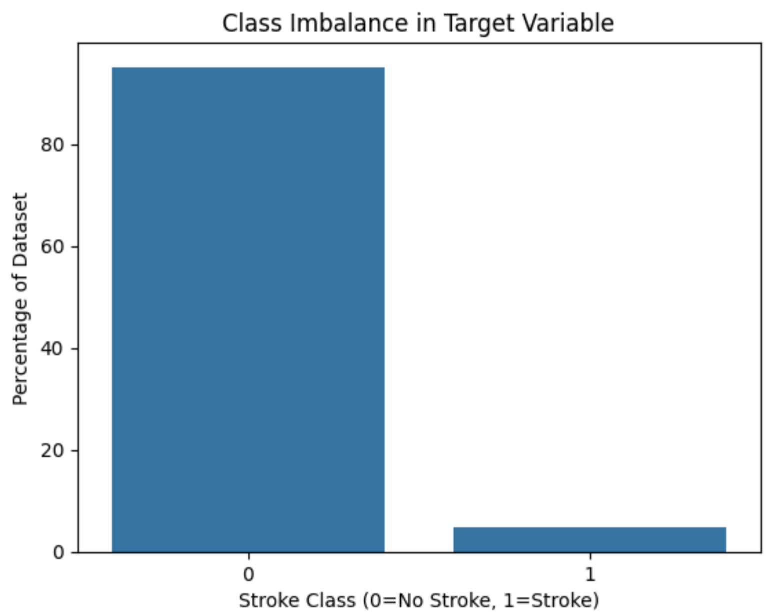

Figure 3: Class imbalance bar chart

2.4 Distribution and Outlier Analysis

Distributions:

- Skewed features: avg_glucose_level, bmi, hypertension, heart_disease
- Most patients are middle-aged; stroke cases increase with age.

Outliers:

- Detected in avg_glucose_level and bmi (visible in box plots)
- Outliers likely represent real medical conditions (diabetes, obesity)

- Decision: Retained outliers; used RobustScaler for normalization

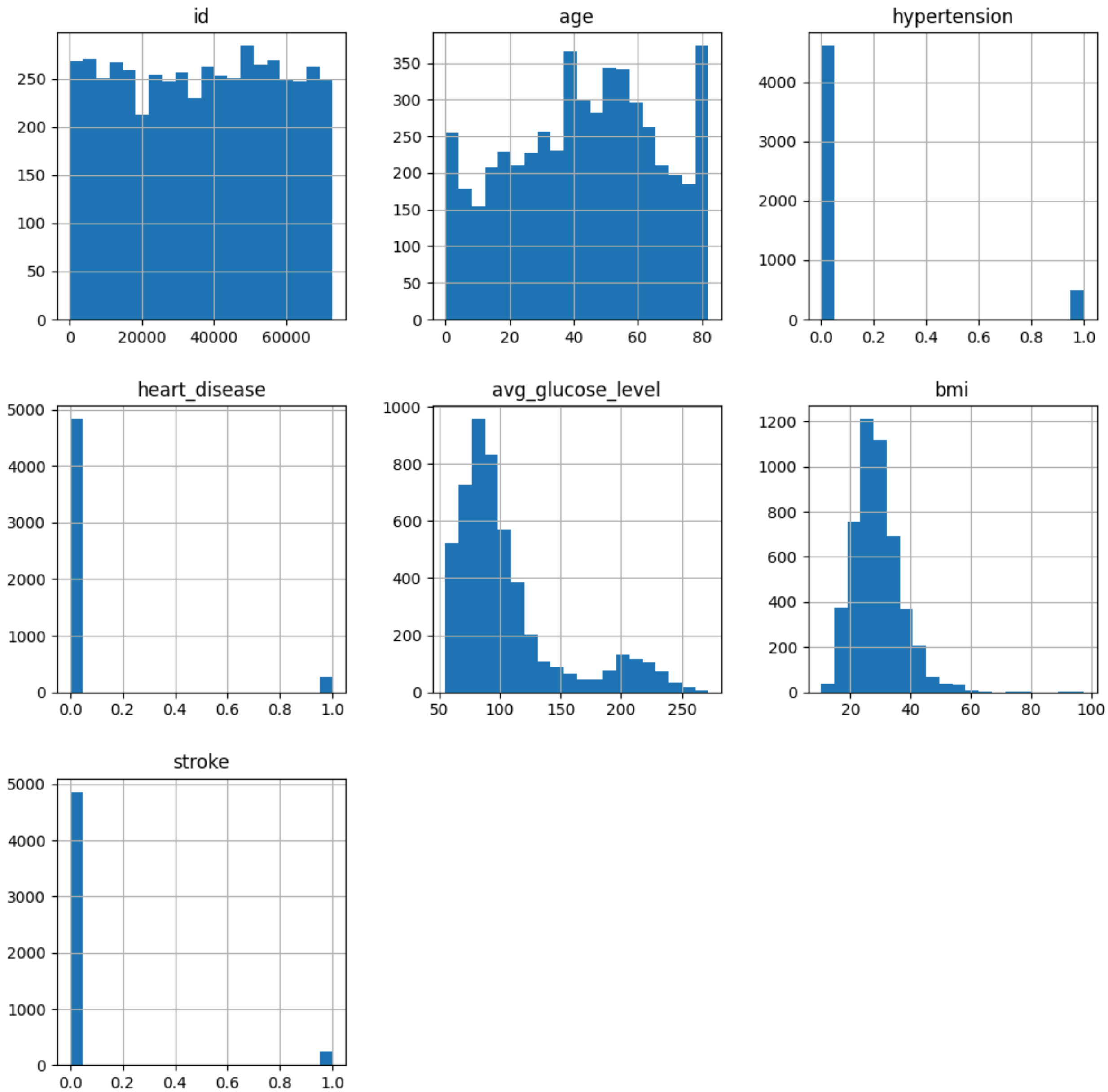

Figure 4: Histograms of numerical features

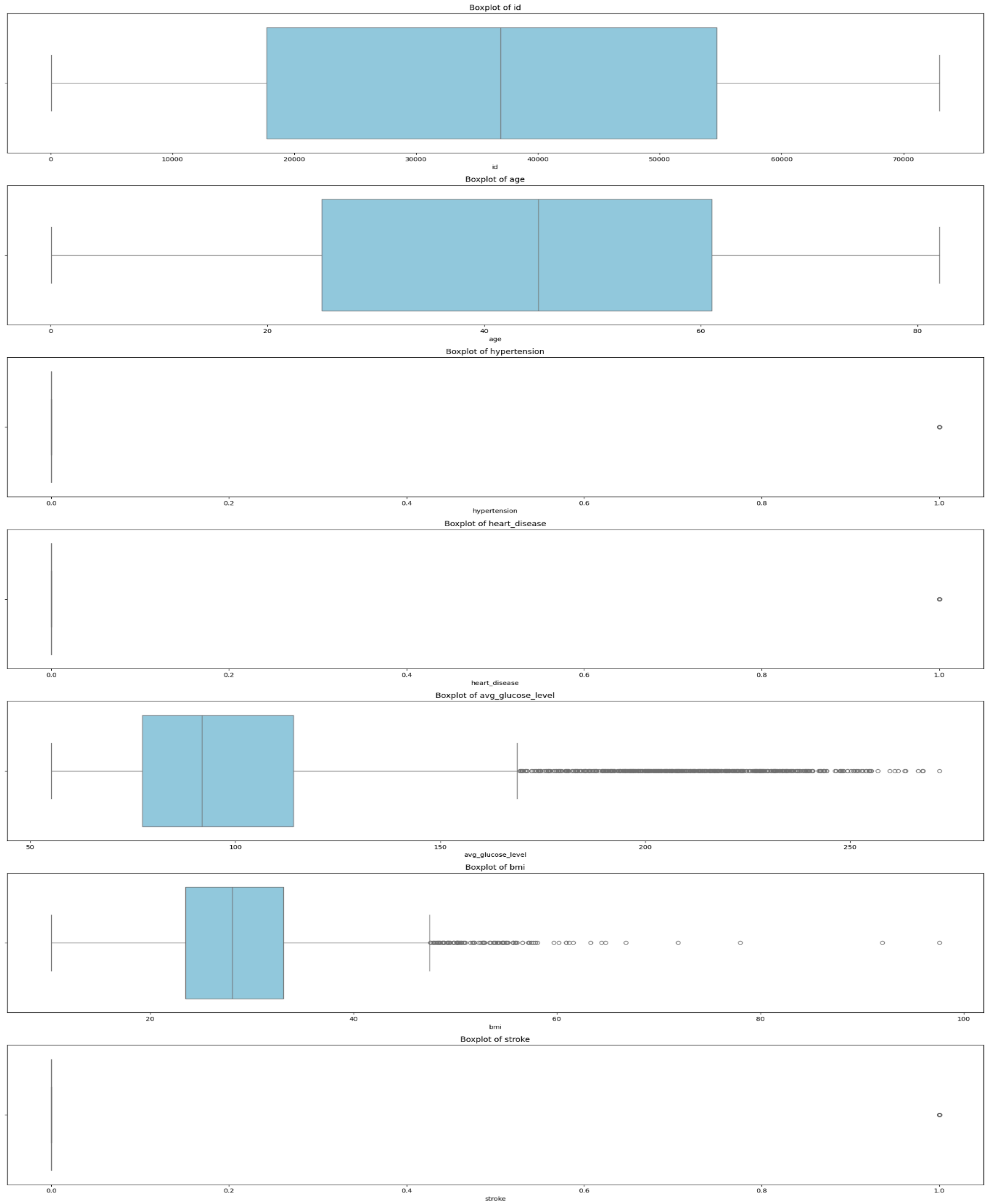

Figure 5: Box plots showing outliers

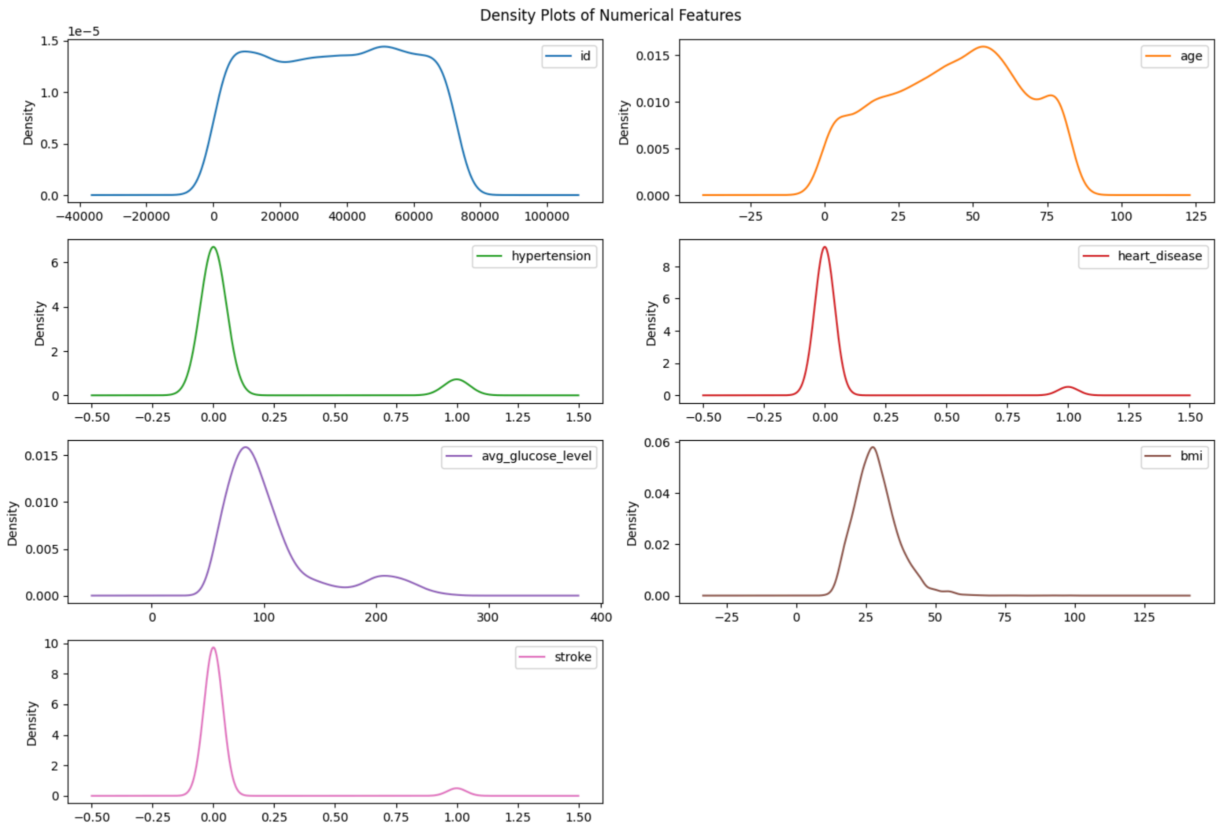

Figure 6: Density plots of numerical features

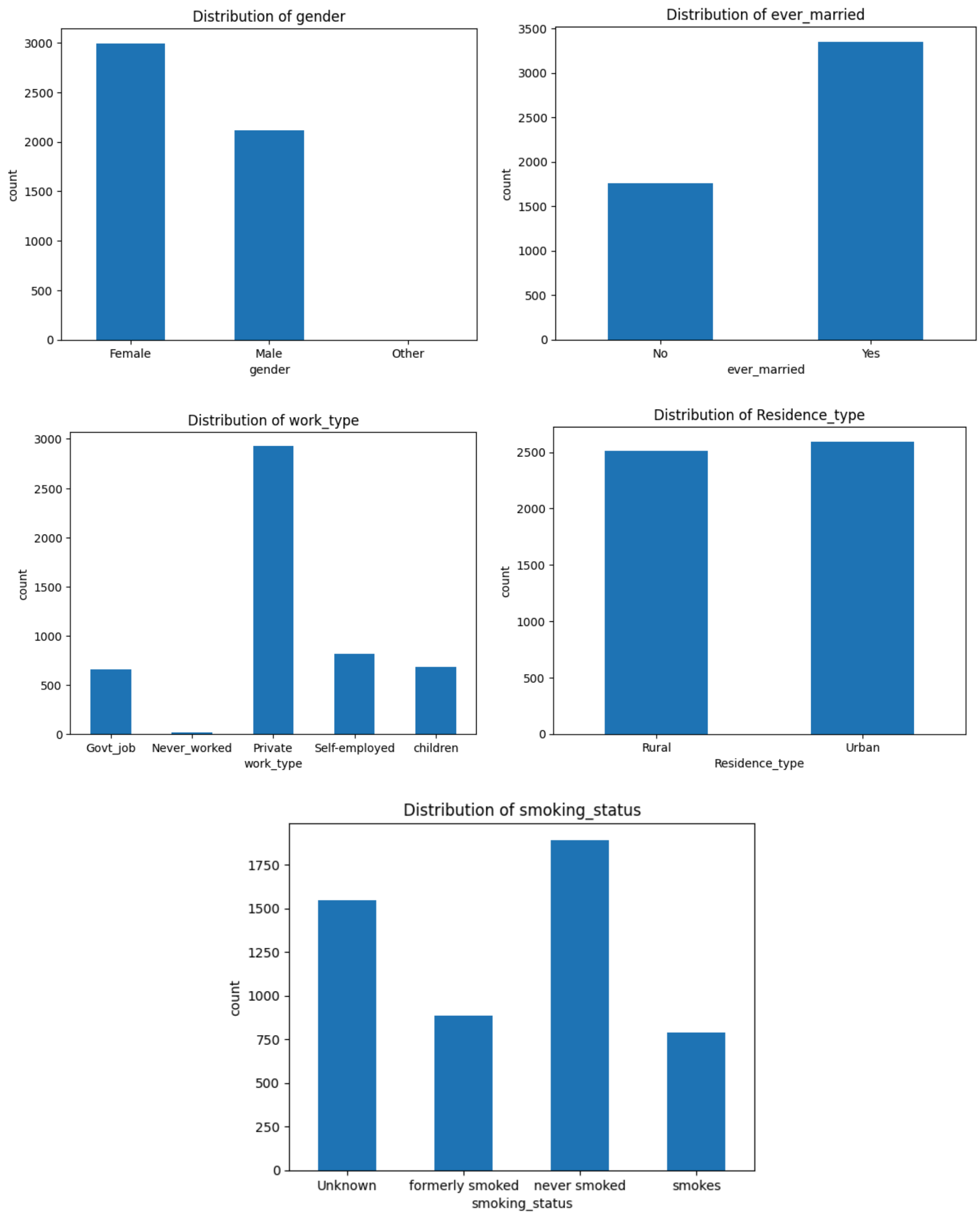

Figure 7: Categorical feature distributions

3. Data Preprocessing & Feature Engineering

3.1 Handling Missing Values

Problem: The BMI column contains missing values
Solution: Imputed missing BMI values using the median
Rationale: Median is robust to outliers (BMI contains outliers)

3.2 Feature Selection

Removed Features:

- `id` - unique identifier with no predictive value
- `ever_married` - weak relationship with stroke risk
- `work_type` - job category, not a direct medical indicator
- `Residence_type` - urban/rural, not strongly correlated

Retained Features (7):

- age, gender, hypertension, heart_disease, avg_glucose_level, bmi, smoking_status

3.3 Categorical Encoding

Categorical variables were converted to numeric codes using label encoding:

Gender	Code
Female	0
Male	1
Other	2
Female	**0**
Male	1
Other	2
Female	**0**
Male	1
Other	2
Male	**1**
Other	2
Other	**2**

Smoking Status	Code
never smoked	0
formerly smoked	1
smokes	2
Unknown	3
never smoked	**0**
formerly smoked	1
smokes	2
Unknown	3
never smoked	**0**
formerly smoked	1
smokes	2
Unknown	3
formerly smoked	**1**
smokes	2
Unknown	3
smokes	**2**
Unknown	3
Unknown	**3**

3.4 Feature Scaling

Scaler Used: RobustScaler
Formula: (X - median) / IQR
Rationale: More resistant to outliers than StandardScaler; critical given outliers in BMI and glucose levels

3.5 Data Stratification

To preserve class distribution (95:5 ratio) in both train and test sets, we used stratified sampling during splitting and applied `class_weight='balanced'` during training where supported.

4. Train-Test Split Strategy

Split Configuration:

- Training Set: 3,577 samples (70%)
- Test Set: 1,533 samples (30%)
- Random State: 5 (for reproducibility)
- Stratification: Enabled (preserves 95:5 class ratio)

Preprocessing Summary:

- Starting samples: 5,110
- After imputation: Complete dataset (BMI filled)
- After encoding: All categorical features converted
- After scaling: Normalized using RobustScaler

5. Model Training & Implementation

Models Trained:

- Logistic Regression - Linear probabilistic classifier; interpretable
- Random Forest - Ensemble of decision trees; handles non-linearity
- Decision Tree - Single tree; prone to overfitting
- K-Nearest Neighbors (KNN) - Instance-based; distance-sensitive
- Naive Bayes - Probabilistic; assumes feature independence
- Neural Network - Deep learning model with 2 hidden layers

Neural Network Architecture:

- Input → Dense(512, ReLU) → Dropout(0.3) → Dense(256, ReLU) → Dense(1, Sigmoid)
- Loss: Binary Crossentropy
- Optimizer: Adam
- Metrics: Accuracy

Training Configuration:

- All models trained on RobustScaler-normalized data
- Class imbalance handling: `class_weight='balanced'` where applicable

6. Model Performance Evaluation

6.1 Initial ROC-AUC Comparison

We evaluated five models using ROC curves. AUC (Area Under Curve) measures discrimination ability: 0.5 = random guessing, 1.0 = perfect classification.

ROC-AUC Scores (5 Models):

Model	AUC Score
Logistic Regression	0.840
Naive Bayes	0.819
Random Forest	0.793
KNN	0.609
Decision Tree	0.527
Logistic Regression	**0.840**
Naive Bayes	0.819
Random Forest	0.793
KNN	0.609
Decision Tree	0.527
Logistic Regression	**0.840**
Naive Bayes	0.819
Random Forest	0.793
KNN	0.609
Decision Tree	0.527
Naive Bayes	**0.819**
Random Forest	0.793
KNN	0.609
Decision Tree	0.527
Random Forest	**0.793**
KNN	0.609
Decision Tree	0.527
KNN	**0.609**
Decision Tree	0.527
Decision Tree	**0.527**

- **Finding: Logistic Regression achieved the highest AUC (0.840), followed by Naive Bayes (0.819) and Random Forest (0.793).**

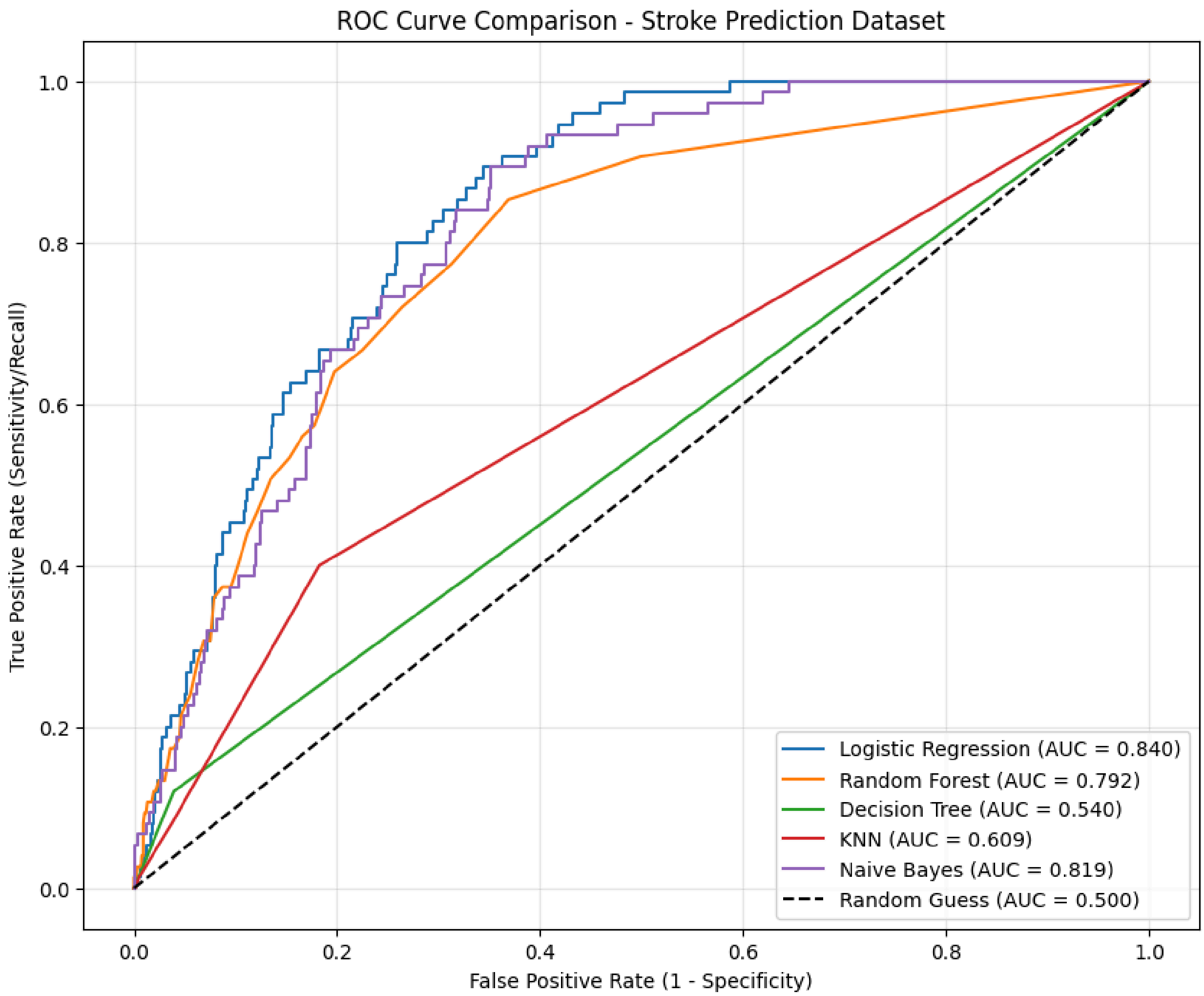

Figure 8: ROC curves for 5 models

6.2 Accuracy Comparison

We selected three top-performing models for detailed evaluation:

Model	Accuracy
Neural Network	94.65%

Random Forest	95.17%
Logistic Regression	74.30%
Neural Network	**94.65%**
Random Forest	95.17%
Logistic Regression	74.30%
Neural Network	**94.65%**
Random Forest	95.17%
Logistic Regression	74.30%
Random Forest	**95.17%**
Logistic Regression	74.30%
Logistic Regression	**74.30%**

Critical Note: Due to severe class imbalance (95% non-stroke), accuracy is misleading. A model predicting all negatives achieves ~95% accuracy but detects zero stroke cases.

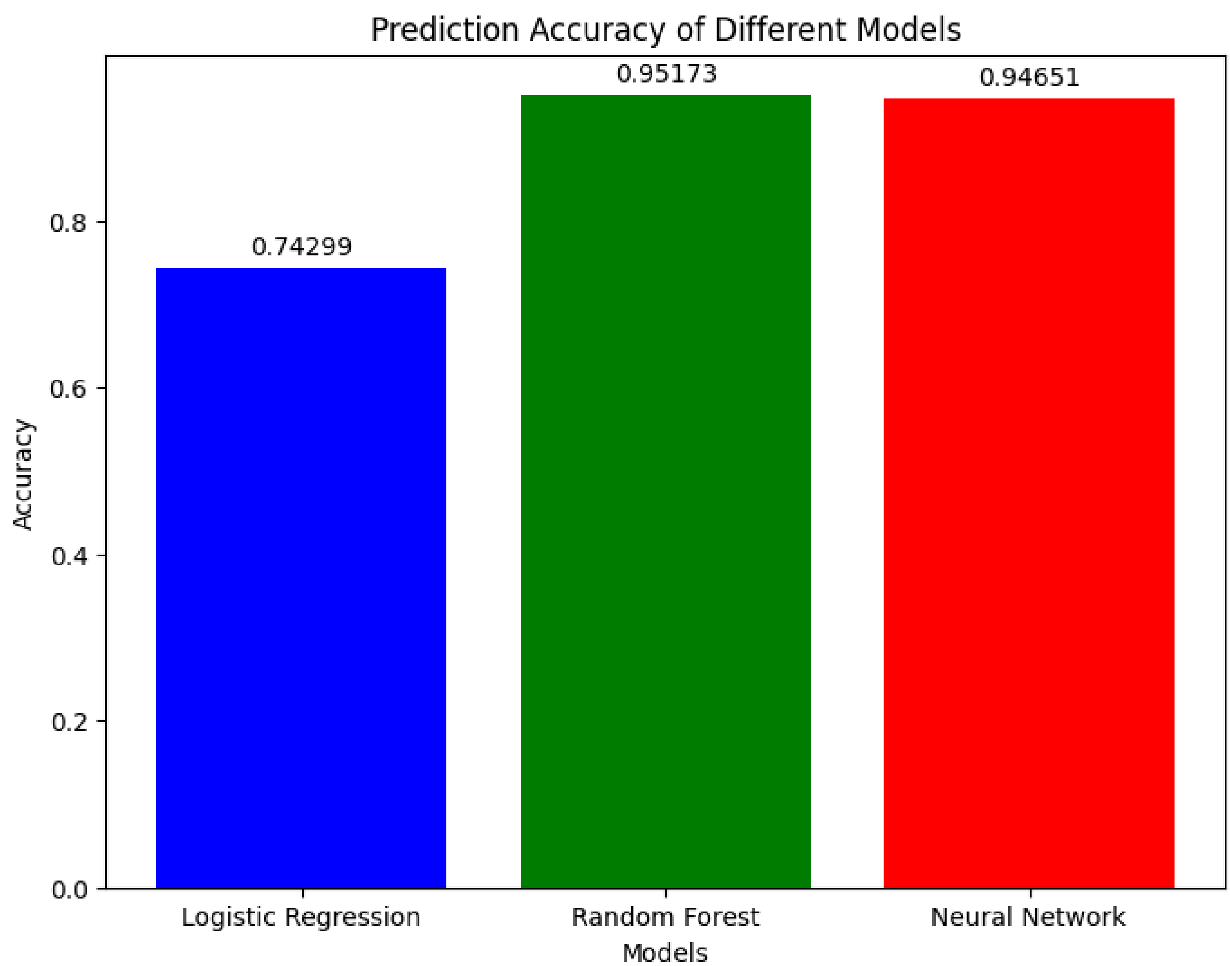

Figure 9: Accuracy comparison bar chart

6.3 Classification Metrics

Precision, Recall, and F1-scores provide deeper insight:

Logistic Regression:

- Class 0 (No Stroke): Precision=0.98, Recall=0.74, F1=0.85
- Class 1 (Stroke): Precision=0.13, Recall=0.77, F1=0.23
- Overall Accuracy: 74%

Random Forest:

- Class 0 (No Stroke): Precision=0.95, Recall=1.00, F1=0.98
- Class 1 (Stroke): Precision=1.00, Recall=0.01, F1=0.03
- Overall Accuracy: 95%

Neural Network:

- Class 0 (No Stroke): Precision=0.95, Recall=1.00, F1=0.97
- Class 1 (Stroke): Precision=0.00, Recall=0.00, F1=0.00
- Overall Accuracy: 95%

Critical Insight: Random Forest and Neural Network achieve high accuracy by predicting mostly/all negatives, resulting in extremely low stroke detection (recall ≤ 0.01).

6.4 Confusion Matrix Analysis

Confusion matrices reveal True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). In medical screening, False Negatives (missed strokes) are critical.

Logistic Regression:

	Predicted No Stroke	**Predicted Stroke**
Actual No Stroke	TN = 1,081	FP = 377
Actual Stroke	FN = 17	TP = 58
Actual No Stroke	**TN = 1,081**	**FP = 377**
Actual Stroke	FN = 17	TP = 58
Actual No Stroke	**TN = 1,081**	**FP = 377**
Actual Stroke	FN = 17	TP = 58
Actual Stroke	**FN = 17**	**TP = 58**

- Stroke Recall: 58/75 = 77%
- Detects most stroke cases but generates many false alarms

Random Forest:

	Predicted No Stroke	**Predicted Stroke**
Actual No Stroke	TN = 1,458	FP = 0
Actual Stroke	FN = 74	TP = 1
Actual No Stroke	**TN = 1,458**	**FP = 0**
Actual Stroke	FN = 74	TP = 1
Actual No Stroke	**TN = 1,458**	**FP = 0**

Actual Stroke	FN = 74	TP = 1
Actual Stroke	**FN = 74**	**TP = 1**

- Stroke Recall: 1/75 = 1.3%
- Misses 74 out of 75 stroke cases (unacceptable for screening)

Neural Network:

	Predicted No Stroke	**Predicted Stroke**
Actual No Stroke	TN = 1,451	FP = 7
Actual Stroke	FN = 75	TP = 0
Actual No Stroke	**TN = 1,451**	**FP = 7**
Actual Stroke	FN = 75	TP = 0
Actual No Stroke	**TN = 1,451**	**FP = 7**
Actual Stroke	FN = 75	TP = 0
Actual Stroke	**FN = 75**	**TP = 0**

- Stroke Recall: 0/75 = 0%
- Predicts nearly all samples as "no stroke" (trivial majority-class solution)

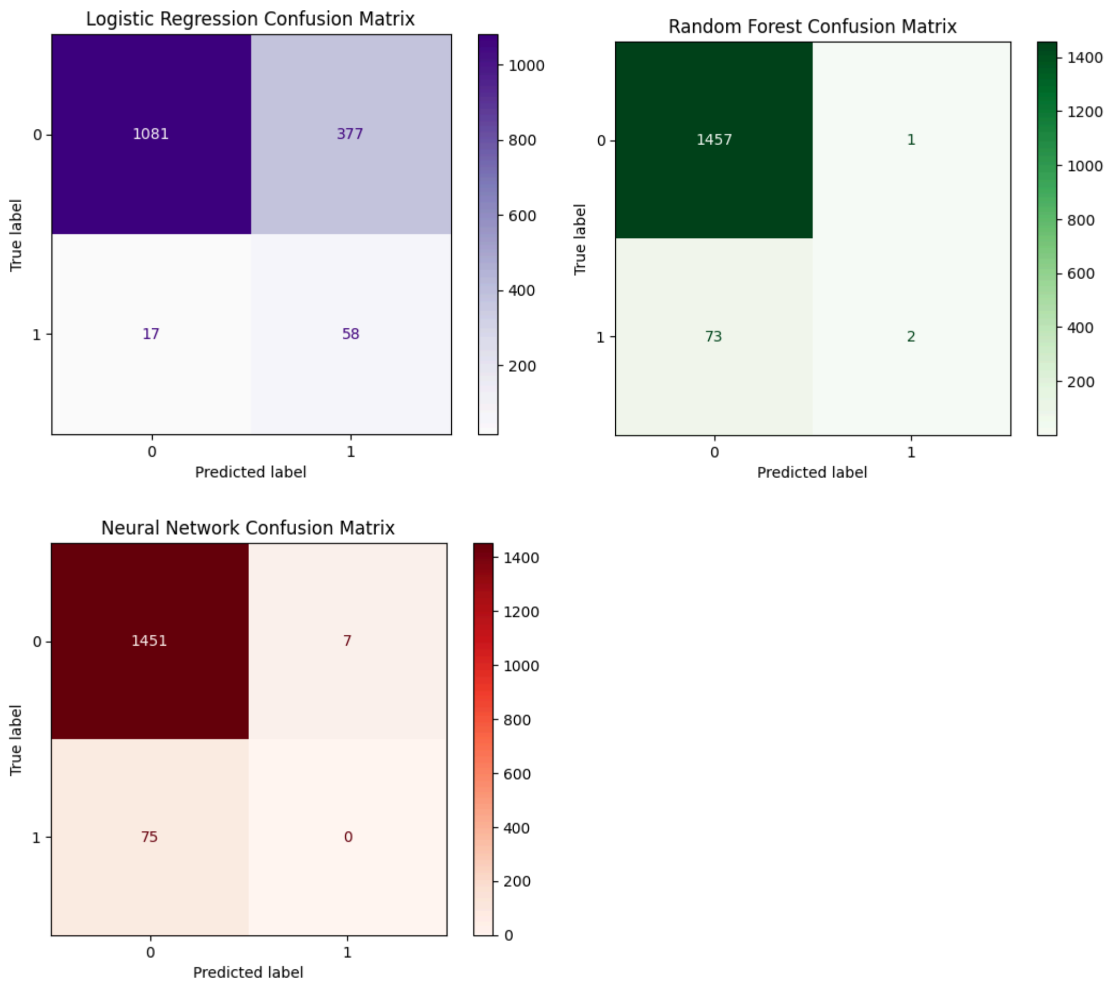

Figure 10: Confusion matrices for all 3 models

6.5 Final ROC-AUC Comparison (3 Models)

Final ROC-AUC scores for the three selected models:

Model	AUC Score
Logistic Regression	0.84
Neural Network	0.82

Random Forest	0.81
Logistic Regression	**0.84**
Neural Network	0.82
Random Forest	0.81
Logistic Regression	**0.84**
Neural Network	0.82
Random Forest	0.81
Neural Network	**0.82**
Random Forest	0.81
Random Forest	**0.81**

Despite similar AUC scores, confusion matrices reveal vastly different behaviors. Logistic Regression achieves the best balance between discrimination (AUC) and stroke detection (recall).

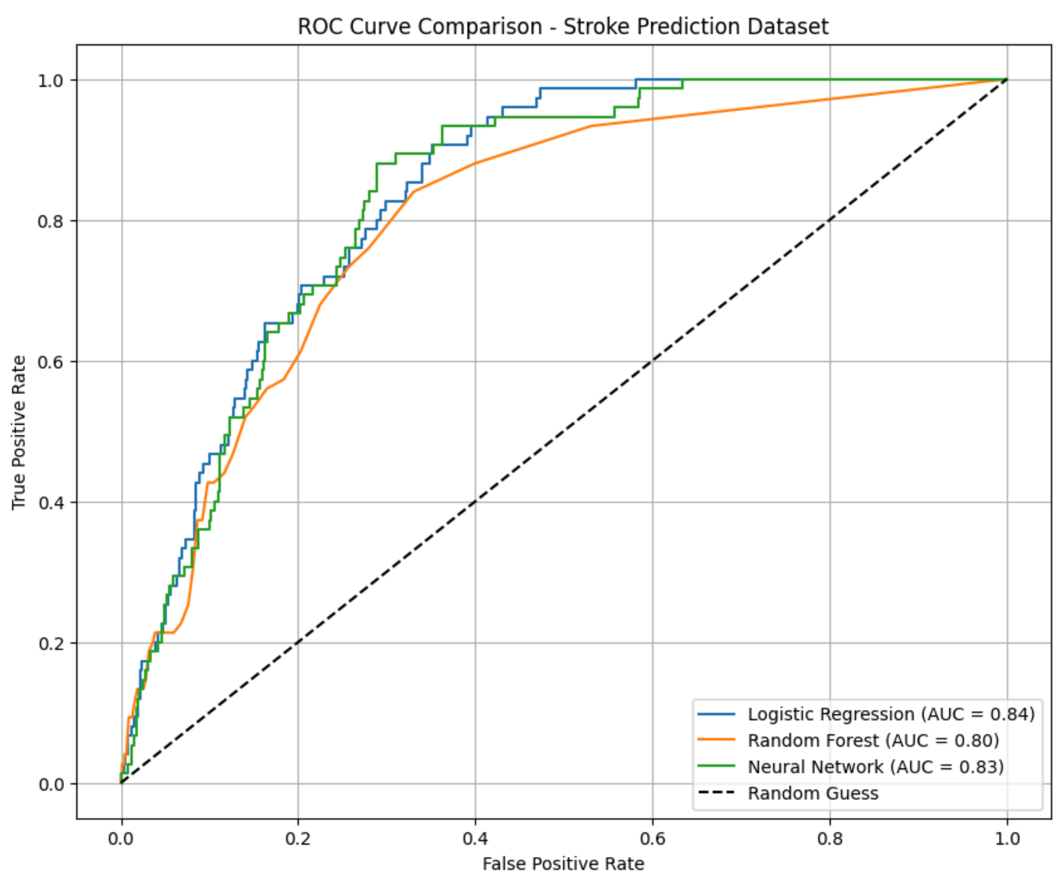

Figure 11: ROC curves for 3 final models

7. Conclusions & Recommendations

Model Performance Summary

Logistic Regression:

- ✓ Best ROC-AUC (0.84)
- ✓ Highest stroke detection (77% recall)
- ✓ Interpretable coefficients
- ✗ Lower accuracy (74.30%)
- ✗ Higher false positives (377)

Random Forest:

- ✓ High accuracy (95.17%)
- ✓ Nearly zero false positives
- ✗ Extremely low stroke recall (1.3%)
- ✗ Misses 74/75 stroke cases

Neural Network:

- ✓ High apparent accuracy (94.65%)
- ✗ Zero stroke detection (0% recall)
- ✗ Predicts the majority class only
- ✗ Not usable in its current form

Best Model Selection

Recommended Model: Logistic Regression

Rationale:

- Best ROC-AUC discrimination (0.84)
- Highest stroke detection rate (77% recall)
- Clinically, false positives are manageable; false negatives are dangerous
- Interpretable coefficients aid clinical understanding

Addressing Limitations

The severe class imbalance (95.1% vs 4.9%) causes models to optimize for accuracy by predicting the majority class, leading to poor stroke detection.

Recommended Improvements:

- Resampling Techniques: Apply SMOTE to increase minority class representation
- Cost-Sensitive Learning: Stronger penalties for false negatives
- Threshold Tuning: Lower decision threshold to increase sensitivity
- Data Collection: Gather more stroke-positive samples
- Cross-Validation: Use stratified k-fold for robust performance estimates
- Ensemble Methods: Combine multiple models with weighted voting

Final Recommendations

For Clinical Deployment:

- Deploy Logistic Regression with threshold tuning
- Set decision threshold to maximize recall (e.g., threshold = 0.3 instead of 0.5)
- Use as a screening aid, not a diagnostic replacement
- Monitor performance and retrain periodically
- Implement clinical workflow integration with clear false-positive handling

Medical Context:

- Prioritize minimizing False Negatives (missed stroke cases)
- False positives lead to additional testing (acceptable)
- False negatives lead to missed intervention (unacceptable)

8. References

Dataset: Healthcare Stroke Data
Link: https://drive.google.com/file/d/18503AUrsLd25Vd-UgQK8IDy2ZlliKQ5g/view

Git Repository: https://github.com/ZeddhD/Stroke-Prediction-Machine-Learning-Model