Stroke Prediction Analysis

Adam Richman & Ava Guy

2025-03-12

Introduction

Stroke is a leading cause of death and disability worldwide, making early detection and prevention crucial. This project aims to analyze the Stroke Prediction Dataset from Kaggle to identify significant risk factors and develop predictive models using logistic regression and Lasso regression.

Why Logistic Regression?

Logistic regression is well-suited for this problem because: - The outcome variable (stroke) is binary (0 = no stroke, 1 = stroke). - Logistic regression provides interpretable coefficients that indicate how each predictor affects the odds of a stroke occurring. - It is robust to small sample sizes and does not assume normality of predictors. - Feature selection techniques like Lasso can be easily applied to improve model performance. - Unlike linear regression, logistic regression naturally models probabilities, making it a better fit for classification tasks.

Data Processing

The dataset under study includes patient details such as gender, age, health conditions (e.g., hypertension, heart disease), smoking status, and more. Each row represents an individual patient, and the target variable is `stroke` (1 = had a stroke, 0 = no stroke).

```r
# load dataset
data <- read.csv("healthcare-dataset-stroke-data.csv")

# remove 'id' column if it exists
data <- data %>% select(-id)

# Check for missing values
sum(is.na(data))
```

```
## [1] 0
```

```r
# Convert categorical variables to factors
data$gender <- factor(data$gender)
data$ever_married <- factor(data$ever_married)
data$work_type <- factor(data$work_type)
data$Residence_type <- factor(data$Residence_type)
data$smoking_status <- factor(data$smoking_status)
data$hypertension <- factor(data$hypertension)
```

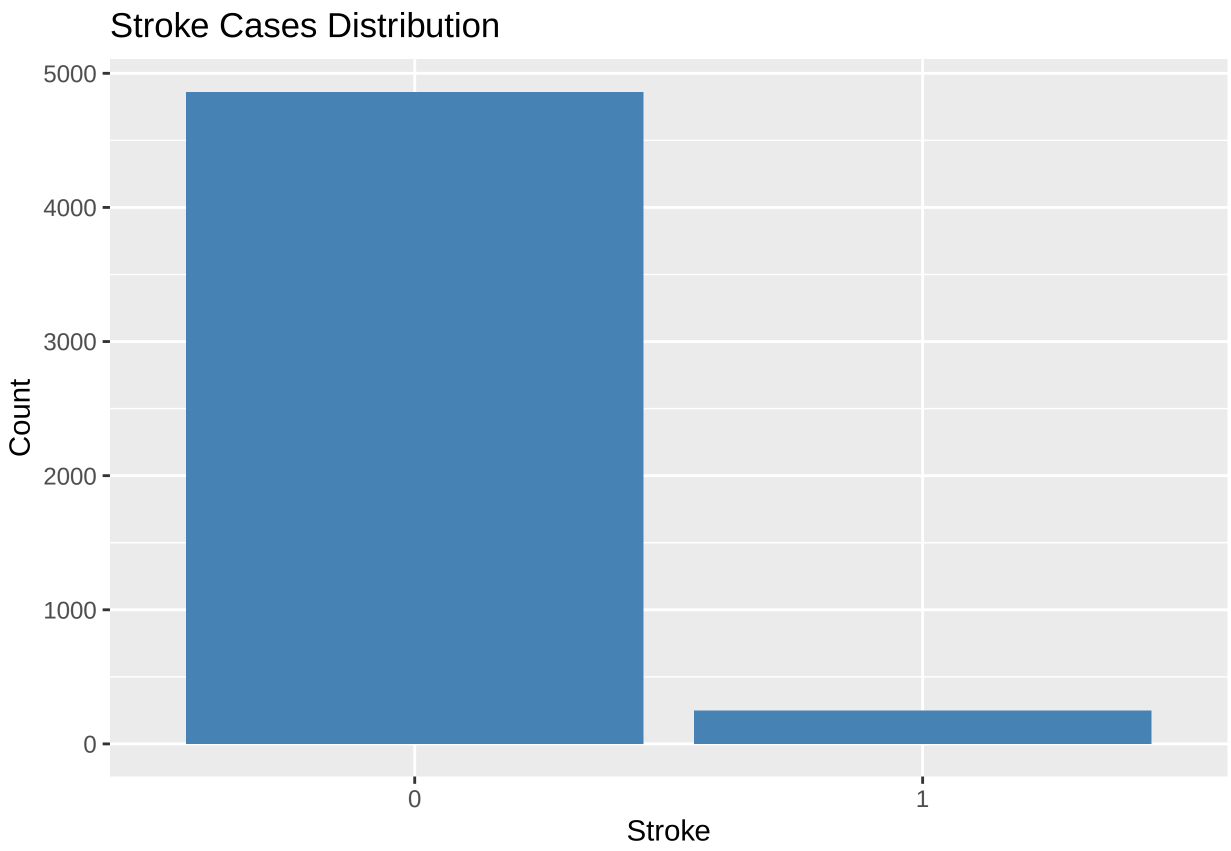
```r
data$heart_disease <- factor(data$heart_disease)
data$stroke <- factor(data$stroke)

# Convert bmi to numeric, treating "N/A" as NA
data$bmi <- as.numeric(gsub("N/A", NA, data$bmi))
```

Note: Although BMI is a numeric variable, it is not strictly continuous. BMI is calculated based on weight and height and has a finite set of values. This makes it discrete, even though it can technically take any real number. Thus, it remains treated as numeric for this project.

Exploratory Data Analysis

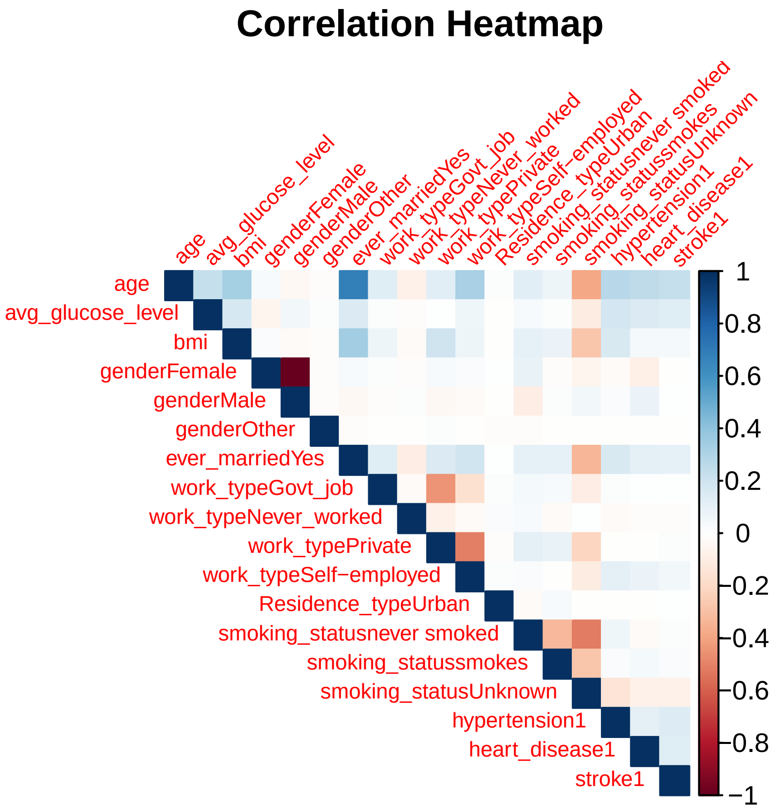
Distribution of Stroke Cases

```r
ggplot(data, aes(x = as.factor(stroke))) +
  geom_bar(fill = "steelblue") +
  labs(title = "Stroke Cases Distribution", x = "Stroke", y = "Count")
```



Correlation Matrix

```r
# data dummies for correlation matrix
data_dummies <- model.matrix(~ gender + ever_married + work_type + Residence_type +
                              smoking_status + hypertension + heart_disease + stroke - 1,
                             data = data)

# combine dummy variables with the original dataset (if needed)
data_full <- cbind(data, data_dummies)

num_data <- data_full %>% select(where(is.numeric))
corr_matrix <- cor(num_data, use = "complete.obs")
corrplot(corr_matrix, method = "color", type = "upper", tl.cex = 0.7, tl.srt = 45, title =
           "Correlation Heatmap", mar = c(1,1,2,2), cl.cex = 0.8)
```



Multicollinearity Check

Multicollinearity can inflate standard errors, making it difficult to determine variable significance. We check for it using the Variance Inflation Factor (VIF):

```r
model <- glm(stroke ~ ., data = data, family = binomial)
vif_values <- vif(model)
print(vif_values)
```

```
##                         GVIF Df GVIF^(1/(2*Df))
## gender            1.044290  2        1.010893
## age               1.375066  1        1.172632
## hypertension      1.065053  1        1.032014
## heart_disease     1.090290  1        1.044170
## ever_married      1.072010  1        1.035379
## work_type         1.287412  4        1.032083
## Residence_type    1.008310  1        1.004146
## avg_glucose_level 1.114002  1        1.055463
## bmi               1.118480  1        1.057582
## smoking_status    1.110560  3        1.017631
```

A VIF > 5 suggests significant multicollinearity, requiring variable removal or transformation. If multicollinearity were present, we would consider removing redundant variables or using principal component analysis (PCA) for dimensionality reduction, but at this time, there does not appear to be multicollinearity, as all VIF values are well below 5.

Train-Validate-Test Split

To ensure reliable model performance, we split the data into: **- 70% training set - 15% validation set - 15% test set**

```r
set.seed(42)
# Create the training set (70% of the data)
trainIndex <- createDataPartition(data$stroke, p = 0.7, list = FALSE)
train <- data[trainIndex, ]

# Remaining data (30%) for validation and test sets
temp <- data[-trainIndex, ]

# Split the remaining data (30%) equally into validation and test sets
valIndex <- createDataPartition(temp$stroke, p = 0.5, list = FALSE)
validation <- temp[valIndex, ]
test <- temp[-valIndex, ]

# Remove rows with missing values from the data
train_clean <- na.omit(train)
test_clean <- na.omit(test)
val_clean <- na.omit(validation)
```

Model and Variable Selection

We use **stepwise selection** and **Lasso regression** to select the best model.

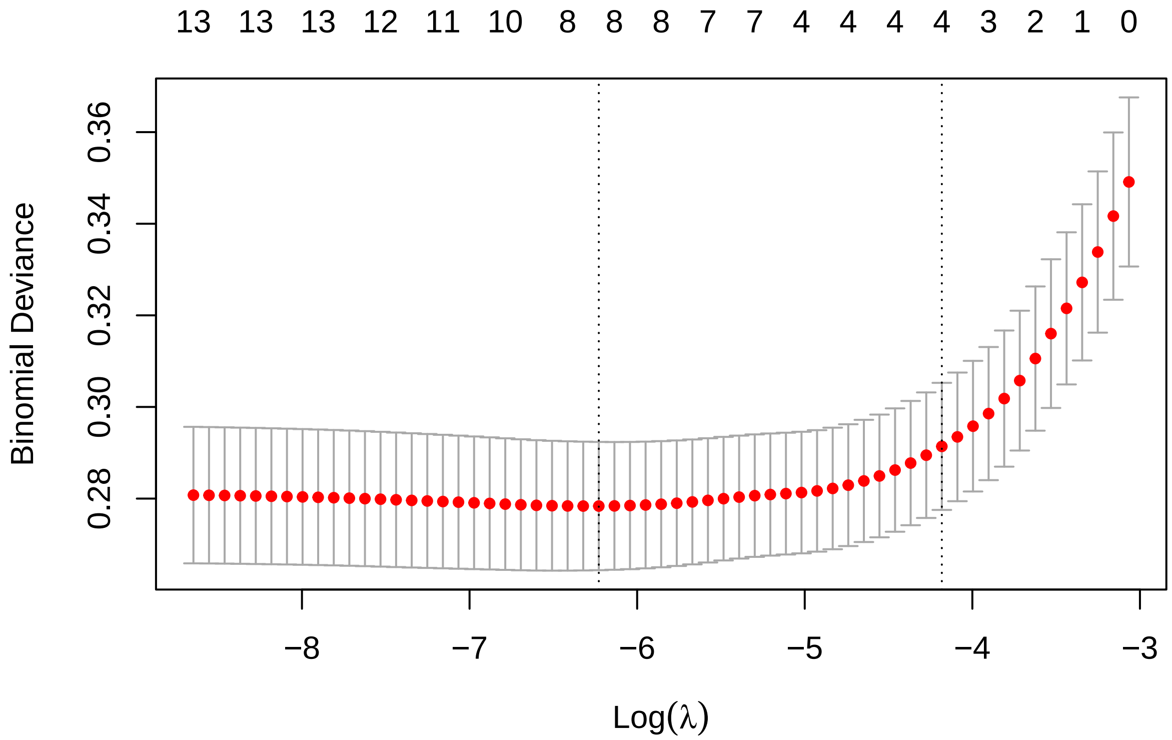
Stepwise Selection

```r
step_model <- step(glm(stroke ~ ., data = train_clean, family = binomial),
                   direction = "both", trace = FALSE)
summary(step_model)
```

```
##
## Call:
## glm(formula = stroke ~ age + hypertension + avg_glucose_level +
##     smoking_status, family = binomial, data = train_clean)
##
## Coefficients:
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -7.990632   0.514428 -15.533  < 2e-16 ***
## age                         0.068617   0.006796  10.096  < 2e-16 ***
## hypertension1               0.477235   0.209479   2.278   0.0227 *
## avg_glucose_level           0.007162   0.001474   4.859 1.18e-06 ***
## smoking_statusnever smoked  0.027059   0.225156   0.120   0.9043
## smoking_statussmokes        0.468441   0.273098   1.715   0.0863 .
## smoking_statusUnknown      -0.378372   0.311873  -1.213   0.2250
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1201.82  on 3436  degrees of freedom
## Residual deviance:  939.99  on 3430  degrees of freedom
## AIC: 953.99
##
## Number of Fisher Scoring iterations: 7
```

Lasso Regression

```r
x <- model.matrix(stroke ~ ., data = train_clean)[, -1]
y <- train_clean$stroke
cv.lasso <- cv.glmnet(x, y, alpha = 1, family = "binomial")
plot(cv.lasso)
```



```r
best_lambda <- cv.lasso$lambda.min
lasso_model <- glmnet(x, y, alpha = 1, lambda = best_lambda, family = "binomial")
coef(lasso_model)
```

```
## 17 x 1 sparse Matrix of class "dgCMatrix"
##                                     s0
## (Intercept)                -7.512670147
## genderMale                  .
## genderOther                 .
## age                         0.063006853
## hypertension1               0.471446587
## heart_disease1              0.311330818
## ever_marriedYes             .
## work_typeGovt_job           .
## work_typeNever_worked       .
## work_typePrivate            0.057349703
## work_typeSelf-employed     -0.244043803
## Residence_typeUrban         .
## avg_glucose_level           0.006532153
## bmi                         .
## smoking_statusnever smoked  .
## smoking_statussmokes        0.266630043
## smoking_statusUnknown      -0.243222653
```

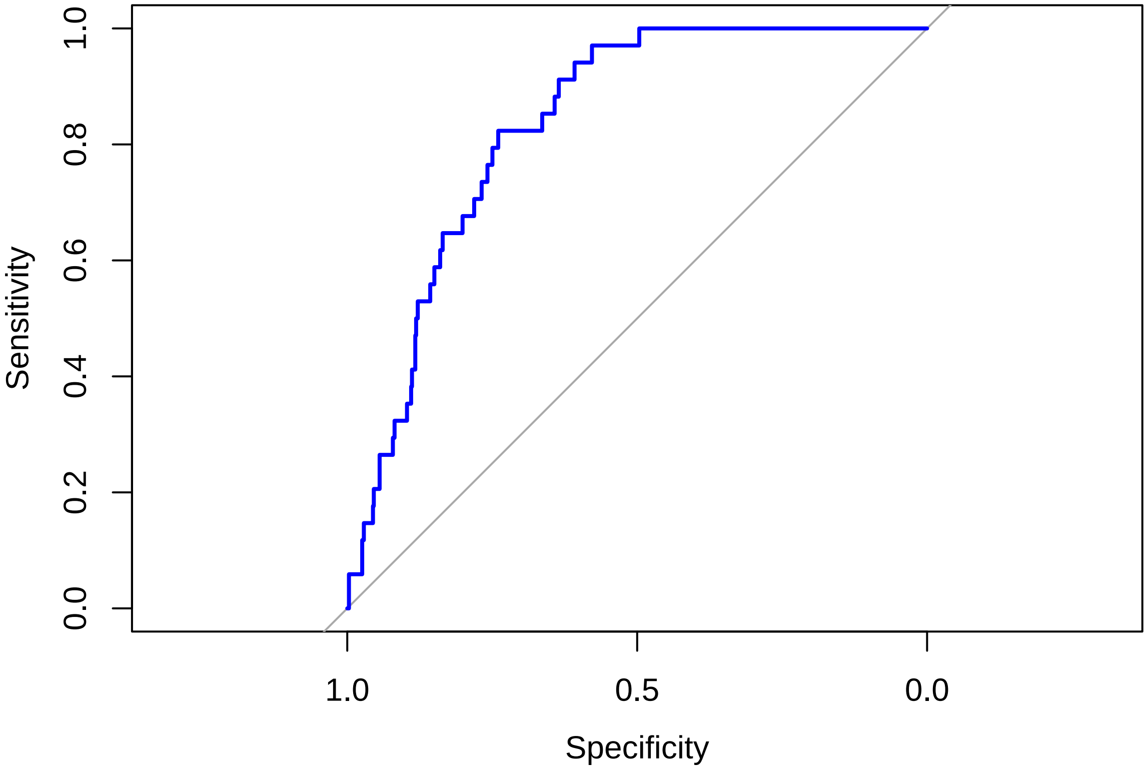
Compared to logistic regression, Lasso regression retained Age, Hypertension, Heart Disease, and Avg. Glucose Level as key predictors, while dropping some categorical features such as Residence Type and Work

Type. This suggests that these dropped variables contribute less predictive power when regularization is applied. The retained features align with clinical expectations, reinforcing their importance in stroke risk prediction.

Model Performance and ROC Curve

We assess model accuracy using precision, recall, and the ROC curve:

```
predictions <- predict(lasso_model, newx = model.matrix(stroke ~ ., test_clean)[, -1],
                       type = "response")
roc_obj <- roc(test_clean$stroke, predictions)
plot(roc_obj, col = "blue")
```



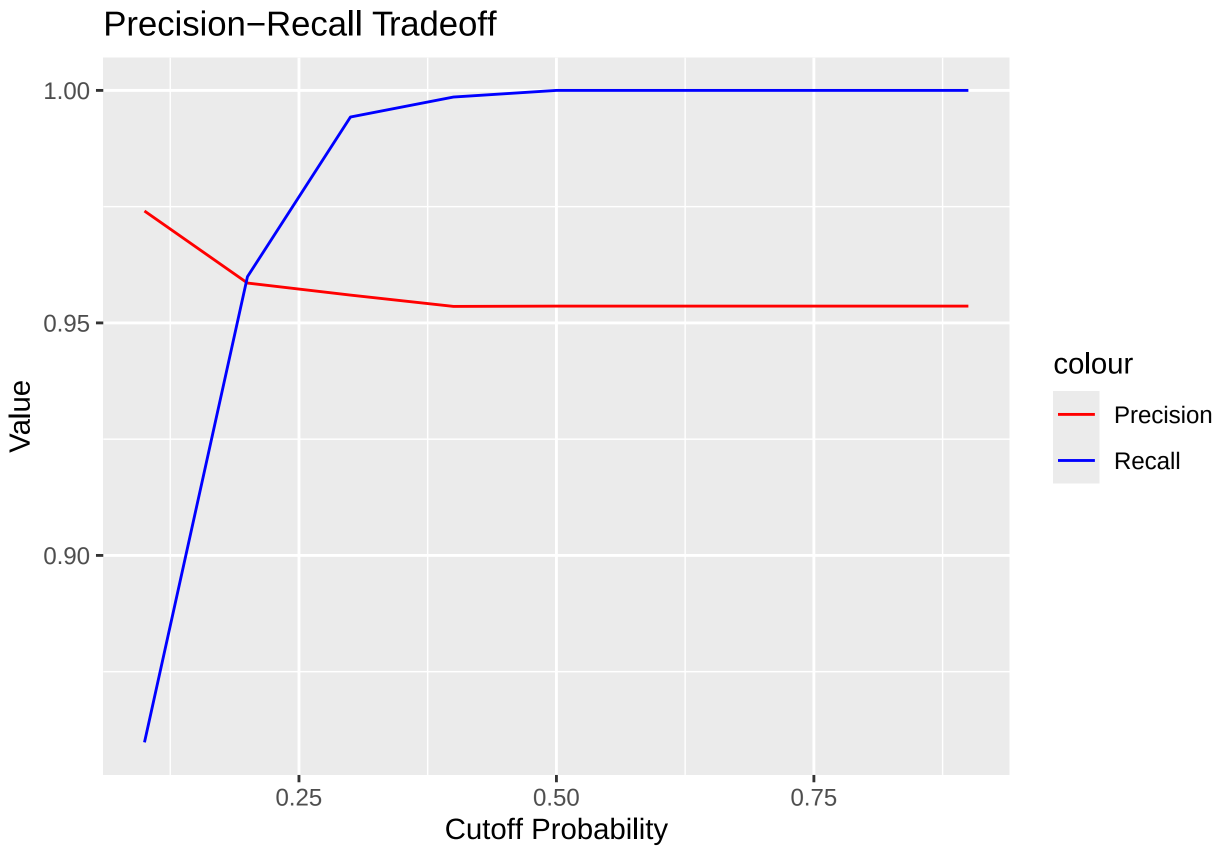
```
auc(roc_obj)
```

```
## Area under the curve: 0.834
```

Cutoff Probability Analysis

Adjusting the probability threshold changes the trade-off between precision and recall:

```r
cutoff_values <- seq(0.1, 0.9, by = 0.1)
results <- data.frame(cutoff = numeric(), precision = numeric(), recall = numeric())

for (cutoff in cutoff_values) {
  predicted_labels <- ifelse(predictions > cutoff, 1, 0)
  conf_matrix <- confusionMatrix(as.factor(predicted_labels),
                                 as.factor(test_clean$stroke))
  precision <- conf_matrix$byClass["Pos Pred Value"]
  recall <- conf_matrix$byClass["Sensitivity"]
  results <- rbind(results, data.frame(cutoff, precision, recall))
}

# Plot precision and recall tradeoff
ggplot(results, aes(x = cutoff)) +
  geom_line(aes(y = precision, color = "Precision")) +
  geom_line(aes(y = recall, color = "Recall")) +
  labs(title = "Precision-Recall Tradeoff", x = "Cutoff Probability", y = "Value") +
  scale_color_manual(values = c("Precision" = "red", "Recall" = "blue"))
```



The cutoff probability threshold in logistic regression determines the probability above which a case is classified as positive (stroke risk) and below which it is classified as negative (no stroke risk). In healthcare, false negatives (missed stroke risks) are more dangerous than false positives, so we prioritize recall over precision to minimize the risk of missing high-risk patients.

The Precision-Recall tradeoff graph shows that at lower cutoff probabilities (e.g., 0.2-0.3), recall is high but precision is lower. Since missing high-risk stroke patients is more dangerous than false positives, a cutoff around 0.3 is preferred. At this threshold, recall remains high while maintaining reasonable precision, making it a clinically sound choice for stroke prediction.

In practical terms, if the model predicts at least a 30% probability of stroke, the patient should be monitored for stroke symptoms and possibly undergo further screening.

Conclusion

By using logistic regression and Lasso feature selection, we identified key predictors of stroke:

- **Age**: Older patients have a significantly higher risk of stroke.
- **Hypertension & Heart Disease**: Both conditions strongly increase stroke likelihood.
- **Smoking Status**: Current smokers are at higher risk, though former smokers also show elevated risk.

Lasso regression retained **age, hypertension, heart disease, and smoking status** as the most predictive variables.

Data Characteristics and Limitations

While BMI is treated as a numeric variable in this analysis, it is technically not continuous. BMI is derived from a finite set of discrete measurements based on weight and height, meaning it cannot take every possible real number value. For example, BMI values are constrained by the possible combinations of weight and height, and while it may appear continuous, it is best considered discrete for analytical purposes. This distinction may affect how the variable interacts with certain models.

Trade-offs in Model Performance

- **Lowering cutoff** increases precision but decreases recall.
- **Raising cutoff** increases recall but reduces precision.
- The ideal threshold depends on whether false positives (unnecessary interventions) or false negatives (missed stroke cases) are more concerning. For overall health, recall is generally regarded as more important, thus a higher cutoff is preferred.

Future Work

While this model effectively identifies stroke risk factors, there are limitations. The dataset is sourced from Kaggle and may not be fully representative of global populations. Certain unmeasured variables, such as genetic predisposition or diet, may also influence stroke risk but are absent from this dataset. Future work could involve using larger, more diverse datasets and incorporating additional health indicators to improve generalizability. Future work could explore:

- More complex machine learning models like Random Forest or Neural Networks.
- Incorporating longitudinal health data for better predictions.
- Addressing class imbalance with resampling techniques.

This analysis provides valuable insights into stroke risk factors, aiding in early intervention strategies.