**Business Understanding**

**Problem Statement**

Stroke remains a leading cause of death and disability worldwide, making early identification of at-risk individuals a critical challenge. The stroke dataset contains demographic and health-related variables which can potentially predict the likelihood of a stroke occurring. Leveraging this data to identify key predictors and risk patterns can greatly enhance preventive strategies and improve patient outcomes.

**Objective**

The objective of this analysis is to utilize the stroke dataset to develop a predictive model that can accurately classify individuals based on their risk of having a stroke. By employing statistical techniques and machine learning algorithms, we aim to identify significant predictors and quantify their impact. This model will help healthcare providers to prioritize interventions and allocate resources more effectively to those at higher risk.

**Data Understanding**

The stroke.df dataset is a comprehensive collection of 5,110 observations across 12 variables, aimed at understanding factors associated with stroke incidence. The dataset comprises both categorical and numerical data types, reflecting a diverse range of demographic and health-related attributes.

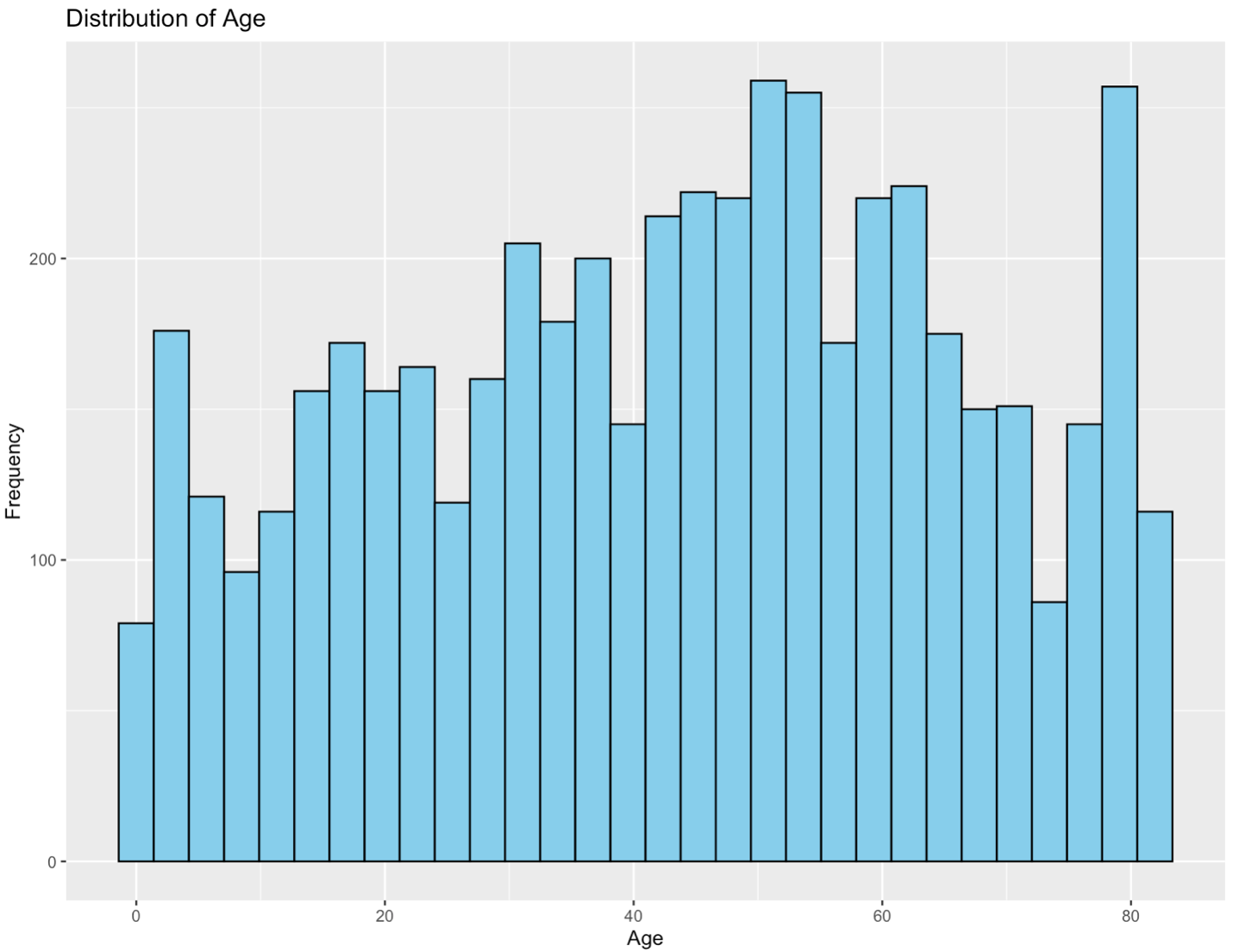
| **Variable** | **Definition** | **Type** | **Constraints / Rules** |
| --- | --- | --- | --- |
| **id** | Unique identifier for each individual | Integer | Must be unique for each individual; no missing values. |
| **gender** | Gender of the individual | Categorical | Possible values: Male, Female, Other; no missing values. |
| **age** | Age of the individual (years) | Numeric | Positive integer, typically ranges from 0 to 100+; no missing values. |
| **hypertension** | Whether the individual has hypertension (high blood pressure) | Binary | Values: 0 (No), 1 (Yes); no missing values. |
| **heart\_disease** | Whether the individual has heart disease | Binary | Values: 0 (No), 1 (Yes); no missing values. |
| **ever\_married** | Whether the individual has ever been married | Categorical | Possible values: Yes, No; no missing values. |
| **work\_type** | Type of employment | Categorical | Possible values: Private, Self-employed, Govt\_job, children, Never\_worked; no missing values. |
| **Residence\_type** | Type of residence (urban or rural) | Categorical | Possible values: Urban, Rural; no missing values. |
| **avg\_glucose\_level** | Average glucose level in blood (mg/dL) | Numeric | Positive value; no missing values. Typical range should be within normal blood glucose levels (e.g., 70–200+ mg/dL). |
| **bmi** | Body Mass Index of the individual | Numeric | Positive value; no missing values. Typical range is 15–50. |
| **smoking\_status** | Smoking status of the individual | Categorical | Possible values: never smoked, formerly smoked, smokes, Unknown. Missing values may be represented as Unknown. |
| **stroke** | Whether the individual has had a stroke | Binary | Values: 0 (No), 1 (Yes); no missing values. |

**Data Preparation**

**Data Review:**

* **Variables**: The dataset includes identifiers (id), categorical variables such as gender, ever\_married, work\_type, Residence\_type, and smoking\_status, as well as numerical variables like age, avg\_glucose\_level, and hypertension. The bmi variable is notably stored as a character type, likely due to the presence of non-numeric entries (e.g., "N/A"), which necessitates preprocessing for numerical analysis.
* **Health Indicators**: Key health indicators present in the dataset are hypertension, heart\_disease, avg\_glucose\_level, and bmi, which are crucial for assessing stroke risk.
* **Target Variable**: The stroke variable is binary, indicating whether an individual has experienced a stroke, making it suitable for binary classification tasks in predictive modeling.

**Data Exploration**

****

The histogram of age distribution in my stroke dataset shows a bimodal distribution, with peaks among young adults in their 20s and older adults in their 60s and 70s. This variation indicates that my dataset encompasses a wide age range, which is critical for analyzing how stroke risk factors and prevalence vary across different life stages.

**Key Insights:**

* **Age-Related Risk**: Observing two prominent age groups underscores the importance of considering age as a crucial factor in stroke risk analysis.
* **Targeted Analysis**: The bimodal distribution suggests that I should consider segmenting further analyses by these age groups. This approach could help uncover age-specific risk factors and outcomes, enhancing the effectiveness of targeted stroke prevention strategies.

This understanding of the age distribution is guiding me to tailor predictive models and interventions more precisely by accounting for how age influences stroke risk.

**A graph showing a level of glucose

Description automatically generated**

The histogram of average glucose levels from my stroke dataset shows a right-skewed distribution with two notable peaks. This indicates that most individuals have glucose levels within a normal range, but there's also a significant number with elevated levels, which are critical for assessing stroke risk.

**Personal Insights:**

* **Risk Factor Emphasis**: The presence of elevated glucose levels in a subset of the population underscores the importance of this variable as a potential stroke risk factor.
* **Targeted Health Strategies**: The secondary peak suggests a need for targeted health interventions to manage and monitor glucose levels in those at higher risk.
* **Next Steps**: I plan to further analyze the correlation between glucose levels and stroke incidence to enhance my understanding and approach to stroke prevention.

This histogram is instrumental in directing my focus towards key health metrics that influence stroke risk.

**A graph with red and black bars

Description automatically generated**

The histogram of BMI from my stroke dataset shows a roughly normal distribution centered around the average BMI, with a peak slightly above the middle of the range. This visualization suggests that most individuals have BMI values within a typical range, with fewer outliers on the higher end.

**Personal Insights:**

* **Health Metrics Emphasis**: The concentration around the middle BMI range indicates that a significant portion of the population maintains a BMI that falls within what's generally considered normal or slightly overweight, according to common health standards.
* **Risk Factor Analysis**: Since BMI can be a risk factor for various health issues, including stroke, observing this distribution helps me understand the general health profile of the population in terms of body weight.
* **Next Steps**: I plan to analyze the relationship between BMI categories and stroke incidence to see if higher BMI values correlate with increased stroke risk.

This histogram provides a clear starting point for more detailed investigations into how body weight might influence stroke risk within this dataset.

A graph of a number of people

Description automatically generated

The bar chart displaying the gender distribution in my stroke dataset reveals a higher number of females compared to males, with a smaller representation of individuals identifying as 'Other'.

**Personal Insights:**

* **Population Representation**: This distribution suggests a predominance of female participants in the dataset, which could be significant if gender-specific factors influence stroke risk.
* **Inclusive Data Collection**: The inclusion of an 'Other' category is notable as it provides insights into the diversity of gender identities among the participants, which is important for comprehensive health studies.

A graph with red and green squares

Description automatically generated

The box plot of age distribution related to stroke incidence shows a clear difference in age profiles between those who have experienced a stroke and those who have not. The median age of individuals who have had a stroke is notably higher than those without a stroke, underscoring age as a significant risk factor.

**Personal Insight:**

* **Age as a Primary Risk Factor**: This visualization highlights that older age groups are more susceptible to strokes, reinforcing the importance of focusing preventive healthcare efforts on older adults.
* **Policy Implication**: I see a compelling need to prioritize health monitoring and stroke prevention strategies for the elderly, based on the evident age-related increase in stroke risk shown in the chart.

A graph of a bar graph

Description automatically generated

The bar chart showcasing the distribution of stroke incidence by work type in my dataset reveals significant differences in stroke prevalence among various employment categories. Most notably, individuals in private employment show a disproportionately higher incidence of stroke compared to other work types like government jobs, self-employed, and especially those who have never worked or are children.

**Personal Insights:**

* **Work-Related Stress**: The high incidence in the private sector might suggest higher stress levels or unhealthy lifestyle factors associated with these jobs, which could contribute to increased stroke risk.
* **Preventive Measures**: This observation pushes me to consider targeted preventive health measures for individuals in high-stress work environments, particularly in the private sector. Health interventions could include stress management programs and regular health screenings.

**Modeling**

**Decision Tree:** I addressed the significant class imbalance in the stroke dataset by using the ROSE package to balance the number of instances between "Stroke" and "No Stroke" classes. This was a crucial step because the original dataset had a disproportionate number of "No Stroke" cases, which could have led to a biased model favoring the majority class.

**Key Steps I Took:**

1. **Data Balancing**: I employed the ROSE package to oversample the minority class ("Stroke"), effectively balancing the dataset. This approach was essential to prevent the decision tree from being biased toward the majority class.
2. **Model Training**: I trained a decision tree model on this balanced dataset, aiming to predict stroke incidence. This model was specifically tuned to handle the balanced data and potentially uncover more subtle patterns related to stroke risks.
3. **Model Evaluation**: I assessed the model's performance using several metrics:
   * **Accuracy**: 77.98%, which indicates a good overall rate of correct predictions.
   * **Precision**: 81.26%, showing a high percentage of predicted stroke cases were actual stroke cases.
   * **Recall (Sensitivity)**: 72.74%, indicating that I captured a substantial proportion of all actual stroke cases.
   * **F1 Score**: 76.76%, a balance between precision and recall, reflecting the model's accuracy and robustness.
   * **Specificity**: 83.23%, demonstrating the model's ability to identify true negatives effectively.
   * **AUC**: 80.75%, reflecting the model's capability to distinguish between the classes across all thresholds.

**Logistic Regression:** The logistic regression model and the decision tree both performed well on the stroke prediction task, but they have distinct strengths as shown by their evaluation metrics:

* **Logistic Regression**:
  + **Accuracy**: 78.40%
  + **Precision**: 80.13%
  + **Recall (Sensitivity)**: 75.51%
  + **F1 Score**: 77.75%
  + **Specificity**: 81.28%
  + **AUC**: 85.04%

**Neural Network:** The neural network model for stroke prediction was evaluated using several metrics:

1. **Accuracy**: **85.91%**
   * The proportion of correct predictions out of the total predictions.
2. **Precision**: **86.43%**
   * The proportion of true stroke predictions among all predicted stroke cases.
3. **Recall (Sensitivity)**: **85.19%**
   * The proportion of actual stroke cases correctly identified by the model.
4. **F1 Score**: **85.80%**
   * The harmonic mean of precision and recall, balancing false positives and false negatives.
5. **Specificity**: **86.63%**
   * The proportion of non-stroke cases correctly identified by the model.
6. **AUC (Area Under the ROC Curve)**: **87.08%**
   * A measure of the model's ability to distinguish between stroke and non-stroke cases across various thresholds.

These results indicate that the neural network is a robust and reliable model for predicting stroke outcomes.

For my primary objective of achieving high accuracy and recall, I found the neural network to be the most optimal technique. It achieved the highest accuracy (85.91%), recall (85.19%), and AUC (87.08%), making it best suited for scenarios where prediction performance is critical, even if interpretability is less important.

For my secondary objective, where interpretability is essential (e.g., identifying key factors driving stroke risks), I would recommend logistic regression. It provides a good balance between performance and interpretability, making it easier to explain predictions.

In resource-constrained scenarios, where computational power or simplicity is a priority, I would opt for a decision tree. It offers reasonable performance and the added advantage of visualization, which makes decision-making more straightforward.

**Utilization of Peer Feedback**

Throughout the course, peer feedback played a pivotal role in improving the quality and outcomes of my project. One key piece of feedback I received in a previous project highlighted the limitations caused by an imbalanced dataset, which prevented my models from achieving higher evaluations. To address this, I incorporated synthetic data using techniques like oversampling, which significantly improved my model's performance by providing balanced representation across classes.

Another important takeaway from peer feedback was the value of visual analysis. My peers mentioned that including visual charts in the project helped them better understand the data and follow the analysis more effectively. Based on this feedback, I prioritized incorporating clear and informative visualizations throughout this project, which enhanced both the interpretability and engagement of the analysis.

These reflections highlight how integrating peer suggestions, such as addressing data imbalance and emphasizing visual storytelling, strengthened my project methodologies and outcomes. This iterative process not only improved the technical quality of the models but also made the project more accessible and impactful.

**Final Project Report Compilation**

**1. Data Visualization and Dimension Reduction**

Data visualization was integral to exploring and understanding the dataset. I created charts and plots to highlight patterns and relationships among key features like age, BMI, glucose levels, and stroke outcomes. These visualizations were not only helpful for my own analysis but also received positive peer feedback for making the project more accessible.

Dimension reduction was used to simplify the dataset and focus on the most significant variables. Techniques like correlation analysis helped identify redundancies, ensuring that only impactful features were included in the models. This step improved both the computational efficiency and interpretability of the models.

**2. Performance and Prediction**

Performance evaluation metrics, including accuracy, precision, recall, F1 score, specificity, and AUC, were consistently used to assess the quality of models. These metrics ensured that I could compare and select the best-performing techniques while addressing the project’s objectives of accuracy and recall.

**3. Classification Techniques**

**CART (Classification and Regression Tree):**

I started with CART for its simplicity and interpretability. The decision tree allowed for clear visualization of how different features influenced stroke outcomes. However, its performance was limited due to the imbalanced dataset. This was a key challenge that I addressed later in the project by generating synthetic data to balance the classes.

**Logistic Regression:**

Logistic regression provided a statistical foundation for understanding the relationship between variables and stroke predictions. Its coefficients offered interpretability, making it a valuable tool for identifying the most influential factors. While its performance was slightly better than CART, logistic regression struggled to capture complex, non-linear relationships in the data.

**Neural Networks:**

Neural networks proved to be the most powerful technique for this project, achieving the highest performance metrics, including an accuracy of 85.91%, recall of 85.19%, and an AUC of 87.08%. The network effectively captured complex patterns in the data, addressing the limitations of simpler models like CART and logistic regression. However, the trade-off was the need for significant computational resources and reduced interpretability.

**4. Cluster Analysis and Text Mining**

Cluster analysis was used to group similar records based on key features, offering insights into underlying patterns in the dataset. While not directly related to the classification task, it provided valuable exploratory insights and supported feature engineering.

Text mining techniques were applied to process and analyze text-based data within the dataset. While this was a supplementary analysis, it showcased how unstructured data could add context to structured datasets.

**Challenges Faced**

1. **Imbalanced Dataset:**
   * The dataset’s class imbalance initially hindered model performance, particularly for CART and logistic regression. Synthetic data generation using oversampling methods significantly improved model results.
2. **Complex Relationships:**
   * Capturing non-linear relationships required advanced techniques like neural networks, which demanded more computational resources and careful parameter tuning.
3. **Interpretability vs. Performance:**
   * Balancing interpretability (e.g., CART and logistic regression) with performance (neural networks) was a recurring theme in the project.

**Insights Gained**

* Addressing data imbalance is critical for improving model performance. The use of synthetic data was a turning point for achieving better results.
* Data visualization and clear communication of findings are essential for making the analysis accessible and impactful.
* Neural networks excel in predictive performance but require significant resources and lack interpretability.
* Simpler models like logistic regression and CART remain valuable for feature importance analysis and decision-making, particularly when computational resources are limited.

| **Technique** | **Accuracy** | **Precision** | **Recall** | **F1 Score** | **Specificity** | **AUC** | **Strengths** | **Weaknesses** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CART** | 77.98% | 81.26% | 72.74% | 76.76% | 83.23% | 80.75% | Easy to interpret, visualizable | Prone to overfitting, limited recall |
| **Logistic Regression** | 78.40% | 80.13% | 75.51% | 77.75% | 81.28% | 85.04% | Interpretable, efficient | Assumes linearity, less robust for complex patterns |
| **Neural Networks** | 85.91% | 86.43% | 85.19% | 85.80% | 86.63% | 87.08% | High performance, captures complexity | Computationally intensive, less interpretable |

**Code**

# Data Understanding

# Data Dimension

dim(stroke.df)

# Head of data

head(stroke.df)

stroke.df

#Data types

str(stroke.df)

library(dplyr)

# Convert bmi from character to numeric, treating "N/A" as NA

stroke.df$bmi <- as.numeric(gsub("N/A", NA, stroke.df$bmi))

# Check for any other columns that need type correction

stroke.df$age <- as.numeric(stroke.df$age)

stroke.df$avg\_glucose\_level <- as.numeric(stroke.df$avg\_glucose\_level)

# Summary to check for missing values

summary(stroke.df)

# Impute missing values for bmi with the median (excluding NA values)

median\_bmi <- median(stroke.df$bmi, na.rm = TRUE)

stroke.df$bmi[is.na(stroke.df$bmi)] <- median\_bmi

#Convert Categorical Variables to Factors

stroke.df$gender <- as.factor(stroke.df$gender)

stroke.df$ever\_married <- as.factor(stroke.df$ever\_married)

stroke.df$work\_type <- as.factor(stroke.df$work\_type)

stroke.df$Residence\_type <- as.factor(stroke.df$Residence\_type)

stroke.df$smoking\_status <- as.factor(stroke.df$smoking\_status)

# Impute missing values for bmi using the median

median\_bmi <- median(stroke.df$bmi, na.rm = TRUE)

stroke.df$bmi[is.na(stroke.df$bmi)] <- median\_bmi

# Verify changes

summary(stroke.df$bmi)

# Plotting histograms for age, avg\_glucose\_level, and bmi

library(ggplot2)

# Age Distribution

ggplot(stroke.df, aes(x=age)) +

geom\_histogram(bins=30, fill="skyblue", color="black") +

ggtitle("Distribution of Age") +

xlab("Age") +

ylab("Frequency")

# Average Glucose Level Distribution

ggplot(stroke.df, aes(x=avg\_glucose\_level)) +

geom\_histogram(bins=30, fill="lightgreen", color="black") +

ggtitle("Distribution of Average Glucose Level") +

xlab("Average Glucose Level") +

ylab("Frequency")

# BMI Distribution

ggplot(stroke.df, aes(x=bmi)) +

geom\_histogram(bins=30, fill="salmon", color="black") +

ggtitle("Distribution of BMI") +

xlab("BMI") +

ylab("Frequency")

# Gender Distribution

ggplot(stroke.df, aes(x=gender)) +

geom\_bar(fill="lightblue") +

ggtitle("Gender Distribution") +

xlab("Gender") +

ylab("Count")

# Smoking Status Distribution

ggplot(stroke.df, aes(x=smoking\_status)) +

geom\_bar(fill="lightgreen") +

ggtitle("Smoking Status Distribution") +

xlab("Smoking Status") +

ylab("Count")

# Work Type Distribution

ggplot(stroke.df, aes(x=work\_type)) +

geom\_bar(fill="pink") +

ggtitle("Work Type Distribution") +

xlab("Work Type") +

ylab("Count")

# Boxplot for BMI

ggplot(stroke.df, aes(x="", y=bmi)) +

geom\_boxplot(fill="tan") +

ggtitle("Boxplot of BMI")

# Boxplot for Average Glucose Level

ggplot(stroke.df, aes(x="", y=avg\_glucose\_level)) +

geom\_boxplot(fill="orange") +

ggtitle("Boxplot of Average Glucose Level")

# Scatter Plot for Age vs. Avg Glucose Level colored by Stroke Outcome

ggplot(stroke.df, aes(x=age, y=avg\_glucose\_level, color=factor(stroke))) +

geom\_point(alpha=0.6) +

ggtitle("Age vs. Average Glucose Level by Stroke Outcome") +

xlab("Age") +

ylab("Average Glucose Level") +

scale\_color\_manual(values=c("red", "green"), labels=c("No Stroke", "Stroke"))

# Box plot of Age vs Stroke

ggplot(stroke.df, aes(x=factor(stroke), y=age, fill=factor(stroke))) +

geom\_boxplot() +

labs(x="Stroke", y="Age", title="Distribution of Age by Stroke Incidence") +

scale\_fill\_manual(values=c("red", "green"), labels=c("Stroke", "No Stroke")) +

theme\_minimal()

stroke.df$stroke <- factor(stroke.df$stroke, levels = c(0, 1), labels = c("No Stroke", "Stroke"))

# Creating a grouped bar chart for Work Type vs. Stroke Incidence

ggplot(stroke.df, aes(x = work\_type, fill = stroke)) +

geom\_bar(position = "dodge") +

labs(x = "Work Type", y = "Count", title = "Distribution of Stroke Incidence by Work Type") +

scale\_fill\_brewer(palette = "Set1", name = "Stroke Incidence") +

theme\_minimal() +

theme(axis.text.x = element\_text(angle = 45, hjust = 1))

table(stroke.df$stroke) # Check balance

# Load necessary libraries

library(rpart)

library(caret)

install.packages("ROSE")

library(ROSE)

# Applying the ROSE method to generate synthetic data

set.seed(123)

# Calculate total needed samples based on the desired balance

total\_strokes <- sum(stroke.df$stroke == "Stroke")

total\_no\_strokes <- nrow(stroke.df) - total\_strokes

desired\_total <- 2 \* max(total\_strokes, total\_no\_strokes)

# Applying ROSE with adjusted N

rose\_data <- ovun.sample(stroke ~ ., data = stroke.df, method = "over", N = desired\_total)$data

# Check the new balance of classes

table(rose\_data$stroke)

index <- createDataPartition(rose\_data$stroke, p = 0.80, list = FALSE)

train\_data\_balanced <- rose\_data[index, ]

test\_data\_balanced <- rose\_data[-index, ]

# Build the decision tree model

tree.model <- rpart(stroke ~ . - id, data = train\_data\_balanced, method = "class",

control = rpart.control(minsplit = 20, minbucket = 7,

maxdepth = 10, usesurrogate = 2, xval =10 ))

# Plot the decision tree

plot(tree.model, main = "Decision Tree for Stroke Prediction on Balanced Data")

text(tree.model, use.n = TRUE)

# Evaluation

library(pROC)

predictions <- predict(tree.model, newdata = test\_data\_balanced, type = "class")

prob\_predictions <- predict(tree.model, newdata = test\_data\_balanced, type = "prob")[, "Stroke"]

conf\_mat <- confusionMatrix(predictions, test\_data\_balanced$stroke)

# Extracting metrics from confusion matrix

accuracy <- conf\_mat$overall['Accuracy']

precision <- conf\_mat$byClass['Precision']

recall <- conf\_mat$byClass['Sensitivity']

f1\_score <- 2 \* (precision \* recall) / (precision + recall)

specificity <- conf\_mat$byClass['Specificity']

# Print metrics

cat("Accuracy:", accuracy, "\n",

"Precision:", precision, "\n",

"Recall (Sensitivity):", recall, "\n",

"F1 Score:", f1\_score, "\n",

"Specificity:", specificity, "\n")

roc\_result <- roc(response = test\_data\_balanced$stroke, predictor = as.numeric(prob\_predictions))

plot(roc\_result, main = "ROC Curve")

auc\_value <- auc(roc\_result)

cat("AUC:", auc\_value, "\n")

# Building Logistic Regression model.

# Convert categorical variables into dummy variables using model.matrix

dummy\_data <- model.matrix(~ gender + work\_type + smoking\_status - 1, data = rose\_data)

# Combine dummy variables with the rest of the dataset

rose\_data <- cbind(rose\_data[, !(names(rose\_data) %in% c("gender", "work\_type", "smoking\_status"))], dummy\_data)

#Removing id

rose\_data <- rose\_data[, !names(rose\_data) %in% c("id")]

# Scaling Data

preProcValues <- preProcess(rose\_data[, c("age", "avg\_glucose\_level", "bmi")], method = c("center", "scale"))

rose\_data\_scaled <- predict(preProcValues, rose\_data)

# Verify the structure of the prepared dataset

str(rose\_data\_scaled)

# Split into training and testing datasets

set.seed(123)

index <- createDataPartition(rose\_data\_scaled$stroke, p = 0.80, list = FALSE)

train\_data <- rose\_data\_scaled[index, ]

test\_data <- rose\_data\_scaled[-index, ]

# Fit the logistic regression model

logistic\_model <- glm(stroke ~ ., data = train\_data, family = binomial())

# View the model summary

summary(logistic\_model)

# Predict probabilities on the test data

predicted\_probs <- predict(logistic\_model, newdata = test\_data, type = "response")

# Convert probabilities to binary predictions

predicted\_class <- ifelse(predicted\_probs > 0.5, "Stroke", "No Stroke")

predicted\_class <- factor(predicted\_class, levels = c("No Stroke", "Stroke"))

# Confusion matrix

conf\_matrix <- confusionMatrix(predicted\_class, test\_data$stroke)

# Print evaluation metrics

accuracy <- conf\_matrix$overall['Accuracy']

precision <- conf\_matrix$byClass['Precision']

recall <- conf\_matrix$byClass['Sensitivity']

f1\_score <- 2 \* (precision \* recall) / (precision + recall) # F1 Score

specificity <- conf\_matrix$byClass['Specificity']

cat("Accuracy:", accuracy, "\n")

cat("Precision:", precision, "\n")

cat("Recall (Sensitivity):", recall, "\n")

cat("F1 Score:", f1\_score, "\n")

cat("Specificity:", specificity, "\n")

# Generate ROC curve and calculate AUC

roc\_curve <- roc(test\_data$stroke, predicted\_probs)

plot(roc\_curve, main = "ROC Curve for Logistic Regression")

auc\_value <- auc(roc\_curve)

cat("AUC:", auc\_value, "\n")

# Building Neural Network

# Install neuralnet package if not already installed

if (!require("neuralnet")) install.packages("neuralnet")

library(neuralnet)

# Convert stroke variable to numeric binary

rose\_data\_scaled$stroke <- ifelse(rose\_data\_scaled$stroke == "Stroke", 1, 0)

rose\_data\_scaled$ever\_married <- ifelse(rose\_data\_scaled$ever\_married == "Yes", 1, 0)

rose\_data\_scaled$Residence\_type <- ifelse(rose\_data\_scaled$Residence\_type == "Urban", 1, 0)

# Normalize all numeric columns except the target variable

numeric\_columns <- names(rose\_data\_scaled)[!names(rose\_data\_scaled) %in% c("stroke")]

rose\_data\_scaled[numeric\_columns] <- scale(rose\_data\_scaled[numeric\_columns])

set.seed(123)

index <- createDataPartition(rose\_data\_scaled$stroke, p = 0.80, list = FALSE)

train\_data <- rose\_data\_scaled[index, ]

test\_data <- rose\_data\_scaled[-index, ]

# Clean column names

names(train\_data) <- make.names(names(train\_data))

names(test\_data) <- make.names(names(test\_data))

# Create the formula dynamically

predictors <- names(train\_data)[!names(train\_data) %in% c("stroke")]

formula <- as.formula(paste("stroke ~", paste(predictors, collapse = " + ")))

print(formula)

# Train the neural network

library(neuralnet)

nn\_model <- neuralnet(formula,

data = train\_data,

hidden = c(5, 3),

linear.output = FALSE,

lifesign = "minimal",

stepmax = 1e+06)

# Plot the neural network

plot(nn\_model)

# Make predictions on the test data

nn\_predictions <- compute(nn\_model, test\_data[, predictors])

# Extract predicted probabilities

predicted\_probs <- nn\_predictions$net.result

# Convert probabilities to binary predictions (threshold = 0.5)

predicted\_class <- ifelse(predicted\_probs > 0.5, 1, 0)

predicted\_class <- as.factor(predicted\_class)

# Confusion matrix

conf\_matrix <- confusionMatrix(predicted\_class, as.factor(test\_data$stroke))

# Print confusion matrix and evaluation metrics

print(conf\_matrix)

# Extract key metrics

accuracy <- conf\_matrix$overall["Accuracy"]

precision <- conf\_matrix$byClass["Precision"]

recall <- conf\_matrix$byClass["Sensitivity"]

f1\_score <- 2 \* (precision \* recall) / (precision + recall)

specificity <- conf\_matrix$byClass["Specificity"]

cat("Accuracy:", accuracy, "\n")

cat("Precision:", precision, "\n")

cat("Recall (Sensitivity):", recall, "\n")

cat("F1 Score:", f1\_score, "\n")

cat("Specificity:", specificity, "\n")

# Generate ROC curve

roc\_curve <- roc(as.numeric(test\_data$stroke), as.numeric(predicted\_probs))

# Plot the ROC curve

plot(roc\_curve, main = "ROC Curve for Neural Network", col = "blue", lwd = 2)

abline(a = 0, b = 1, col = "gray", lty = 2)

# Calculate AUC

auc\_value <- auc(roc\_curve)

cat("AUC:", auc\_value, "\n")