Healthcare Stroke

MATH 40024/50024: Computational Statistics

December 10, 2023

ACADEMIC INTEGRITY: Every student should complete the project by their own. A project report having high degree of similarity with work by any other student, or with any other document (e.g., found online) is considered plagiarism, and will not be accepted. The minimal consequence is that the student will receive the project score of 0, and the best possible overall course grade will be D. Additional consequences are described at http://www.kent.edu/policyreg/administrative-policy-regarding-student-cheating-and-plagiarism and will be strictly enforced.

Instruction

Goal: The goal of the project is to go through the complete data analysis workflow to answer questions about your chosen topic using a real-life dataset. You will need to acquire the data, munge and explore the data, perform statistical analysis, and communicate the results.

Report: Use this Rmd file as a template. Edit the file by adding your project title in the YAML, and including necessary information in the four sections: (1) Introduction, (2) Computational Methods, (3) Data Analysis and Results, and (4) Conclusion.

Submission: Please submit your project report as a PDF file (8-10 pages, flexible) to Canvas by **11:59 p.m. on December 10**. The PDF file should be generated by "knitting" the Rmd file. You may choose to first generate an HTML file (by changing the output format in the YAML to `output: html_document`) and then convert it to PDF. **20 points will be deducted if the submitted files are in wrong format.**

Grade: The project will be graded based on your ability to (1) recognize and define research questions suitable for data-driven, computational approaches, (2) use computational methods to analyze data, (3) appropriately document the process (with R code) and clearly present the results, and (4) draw valid conclusions supported by the data analysis.

Example topics:

- Post-Hurricane Vital Statistics
- Tidy Tuesday

Datasets: I suggest to work on a dataset with at least thousands of observations and dozens of variables. You may consider (but are not restricted) to use the following data repositories: Data.gov, Kaggle, FiveThirtyEight, ProPublica, and UCI Machine Learning Repository

Introduction [15 points]

1) What research question(s) would you like to answer?

1) Can we build a regression model to estimate the impact of specific demographics (age, gender)/health (hypertension, heart disease)/lifestyle (smoking status, BMI, average glucose level) factors have the strongest correlation with the likelihood of stroke occurrence?

2) Why a data-driven, computational approach may be useful to answer the questions?

A data-driven, computational approach is very beneficial for answering the question regarding the estimation of the impact of various demographic, health, and lifestyle factors on the likelihood of stroke occurrence-

1) Handling Large Dataset: Healthcare stroke dataset often contain a vast amount of information, including numerous demographics, health and lifestyle factors. Computational approaches enable the study of several feature combinations and the identification of the most influential elements leading to stroke risk.
2) Visual Representation: Computational techniques are used for visual representations such as box plot, bar plots, or histogram. These visualizations helps to compare attribute distributions across individuals who have and have not had a stroke. These graphics helps in understanding the distributions of demographics/health/lifestyle parameters.
3) Analysis Complexity: A computational method enables the investigation of complex interactions between numerous demographics/health/lifestyle variables and their influence on stroke occurrence.Regression model can measure the impact of these characteristics and provides numerical measures such as coefficients, p-values to determine the strongest correlation with stroke occurrence.
4) Estimating Model Robustness: Computational approaches such as bootstrapping may be used to evaluate the stability and robustness of regression model used for stroke prediction. It is feasible to measure the variability in model prediction and evaluate the uncertainty associated with the estimated impact of demographics/health/lifestyle variables on stroke dataset.

3) Describe the dataset that you choose.

This is a real-life dataset in World Health Organization (WHO),it is observed that stroke is the 2nd leading cause of death globally, responsible for approximately 11% of total deaths. ID, Gender, Age, Hypertension, Heart disease, ever married, work type, residence type, average glucose level, BMI, smoking status, stroke are prominently featured in the dataset. These columns contain detailed information on passenger demographics, travel details, flight routes, crew information, and flight statuses. This dataset is used to predict whether a patient is likely to get stroke based on the input parameters like gender, age, various diseases, and smoking status. Each row in the data provides relevant information about the patient. This information may be used for research purpose to analyze correlations between different factors and stroke occurrence, building predictive models using computational methods like regression, uncertainty, and data visualization, and understanding the impact of various demographics, health and lifestyle indicators on stroke occurrence.

Computational Methods [30 points]

1) For the choosen dataset, what are the necessary data wrangling steps to make the data ready for subsequent analyses?

Some necessary data wrangling steps to make the data tidy are-

1) Loading the dataset: The stroke dataset is loaded from a CSV file into a data frame called data. Libraries dplyr and tidyr are loaded for data manipulation.

2) Handling Missing Values: Searching for any missing values in each column of the dataset. Examine the dataset for missing, depending on the importance and impact of missing data, imputation with mean/median for numerical columns or mode for categorical columns. Numeric columns bmi and avg_glucose_level that contain missing values are replaced with the mean.

3) Encoding Categorical Variables to factors: Converting the categorical variables like Gender, Ever Married, Type of Work, Residence Type, and Smoking Status into numerical format using label encoding. Categorical columns like gender are converted into factor variables.

4) Feature Scaling: Scaling numerical characteristics like Age, Average Glucose Level, and BMI to a comparable range, as using model sensitive to feature scale i.e. logistic regression. A new BMICategory column is created that bins the bmi into categories like Underweight, Normal.

5) Handling Outliers: Outliers in the dataset must be identified and handled in numerical columns. Detecting if outliers should be removed, transformed, or used with robust modeling approaches that are less susceptible to outliers. Outliers are detected and removed in bmi, age and avg_glucose_level columns.

6) Splitting the data: To evaluate the model, dividing the dataset into training and testing sets. The training set is used to train the model, whereas the testing set is used to evaluate model performance on previously unknown data. The data is split randomly into 80% train and 20% test sets using sample.

7)Printing the dataset and structure after performing all the data wrangling steps.

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)

#Step 1: Loading the data
data<- read.csv("D:/Computational Statistics/Project/healthcare-dataset-stroke-data.csv")

#Step 2: Handling Missing Values(Checking for missing values)
missing_values <- colSums(is.na(data))

#Replacing missing values in numerical columns with mean
data$bmi<- as.numeric(data$bmi)
```

```
## Warning: NAs introduced by coercion
```

```r
data$avg_glucose_level<- as.numeric(data$avg_glucose_level)

data$bmi[is.na(data$bmi)]<- mean(data$bmi, na.rm = TRUE)
data$avg_glucose_level[is.na(data$avg_glucose_level)]<- mean(data$avg_glucose_level,
                                                             na.rm = TRUE)

#Step 3: Encoding Categorical Variables to factors
data$gender<- as.factor(data$gender)
data$ever_married<- as.factor(data$ever_married)
data$work_type<- as.factor(data$work_type)
data$Residence_type<- as.factor(data$Residence_type)
data$smoking_status<- as.factor(data$smoking_status)

#Step 4: Feature Scaling(Creating BMI categories)
data$BMICategory<- ifelse(data$bmi<18.5, "Underweight",
                   ifelse(data$bmi>=18.5 & data$bmi<25, "Normal",
                   ifelse(data$bmi>=25 & data$bmi<30, "Overweight", "Obese")))

#Step 5: Outlier Handling
#Detecting and removing outliers in BMI column
Q1_bmi<- quantile(data$bmi, 0.25, na.rm = TRUE)
Q3_bmi<- quantile(data$bmi, 0.75, na.rm = TRUE)
IQR_value<- Q3_bmi - Q1_bmi
```

```r
data<- data |>
  filter(!(bmi<(Q1_bmi - 1.5 * IQR_value) | bmi>(Q3_bmi + 1.5 * IQR_value)))

#Detecting and removing outliers in age column
Q1_age<- quantile(data$age, 0.25, na.rm = TRUE)
Q3_age<- quantile(data$age, 0.75, na.rm = TRUE)
IQR_value <- Q3_age - Q1_age
data<- data |>
  filter(!(age<(Q1_age - 1.5 * IQR_value) | age>(Q3_age + 1.5 * IQR_value)))

#Detecting and removing outliers in average glucose level column
Q1_glucose<- quantile(data$avg_glucose_level, 0.25, na.rm = TRUE)
Q3_glucose<- quantile(data$avg_glucose_level, 0.75, na.rm = TRUE)
IQR_value_glucose <- Q3_glucose - Q1_glucose
data<- data |>
  filter(!(avg_glucose_level<(Q1_glucose - 1.5 * IQR_value) |
             avg_glucose_level>(Q3_glucose + 1.5 * IQR_value)))


#Step 6: Splitting the Data
set.seed(123)
train_indices<- sample(1:nrow(data), 0.8 * nrow(data))
train_data<- data[train_indices, ]
test_data<- data[-train_indices, ]

#Step 7: Printing the tidy dataset
head(data)
```

```
##      id gender age hypertension heart_disease ever_married work_type
## 1 31112   Male  80            0             1          Yes   Private
## 2 53882   Male  74            1             1          Yes   Private
## 3 10434 Female  69            0             0           No   Private
## 4 27419 Female  59            0             0          Yes   Private
## 5 60491 Female  78            0             0          Yes   Private
## 6 12109 Female  81            1             0          Yes   Private
##   Residence_type avg_glucose_level      bmi smoking_status stroke BMICategory
## 1          Rural            105.92 32.50000   never smoked      1       Obese
## 2          Rural             70.09 27.40000   never smoked      1  Overweight
## 3          Urban             94.39 22.80000   never smoked      1      Normal
## 4          Rural             76.15 28.89324        Unknown      1  Overweight
## 5          Urban             58.57 24.20000        Unknown      1      Normal
## 6          Rural             80.43 29.70000   never smoked      1  Overweight
```

```
str(data)
```

```
## 'data.frame':    4388 obs. of  13 variables:
##  $ id               : int  31112 53882 10434 27419 60491 12109 12095 12175 58202 27458 ...
##  $ gender           : Factor w/ 3 levels "Female","Male",..: 2 2 1 1 1 1 1 1 1 1 ...
##  $ age              : num  80 74 69 59 78 81 61 54 50 60 ...
##  $ hypertension     : int  0 1 0 0 0 1 0 0 1 0 ...
##  $ heart_disease    : int  1 1 0 0 0 0 1 0 0 0 ...
##  $ ever_married     : Factor w/ 2 levels "No","Yes": 2 2 1 2 2 2 2 2 2 1 ...
##  $ work_type        : Factor w/ 5 levels "children","Govt_job",..: 4 4 4 4 4 4 2 4 5 4 ...
##  $ Residence_type   : Factor w/ 2 levels "Rural","Urban": 1 1 2 1 2 1 1 2 1 2 ...
##  $ avg_glucose_level: num  105.9 70.1 94.4 76.2 58.6 ...
##  $ bmi              : num  32.5 27.4 22.8 28.9 24.2 ...
##  $ smoking_status   : Factor w/ 4 levels "formerly smoked",..: 2 2 2 4 4 2 3 3 2 2 ...
##  $ stroke           : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ BMICategory      : chr  "Obese" "Overweight" "Normal" "Overweight" ...
```

2) What exploratory analyses and modeling techniques can be used to answer the research questions?

1) Correlation Analysis: Calculating correlation coefficients between each factor such as age, hypertension, heart disease, average glucose level, bmi to stroke occurrence and identifying factors showing strong correlations coefficients with the likelihood of stroke risk.

2) Attribute distribution Visual Representation: Creating data visualizations using histograms for numerical data and box plots for categorical variables. Bar plot for gender and stroke occurrence, box plot for age and stroke occurrence, histograms for smoking status and stroke occurrence, bar plot for heart Disease, hypertension, and stroke risk, box plot for Glucose Level vs. BMICategory with Stroke Occurrence and comparing these distribution patterns which helps in identifying differences or similarities between different factors and stroke occurrence.

3) Impact Estimation Using Regression Modeling: As stroke risk is binary (1 for stroke occurrence, 0 for no stroke) logistic regression model is performed with demographics/health/lifestyle factors such as age, BMI, hypertension, heart disease as independent variables and stroke occurrence as the dependent variable. Analyzing the coefficients to determine which factors have the greatest influence on stroke risk.

Correlation Analysis

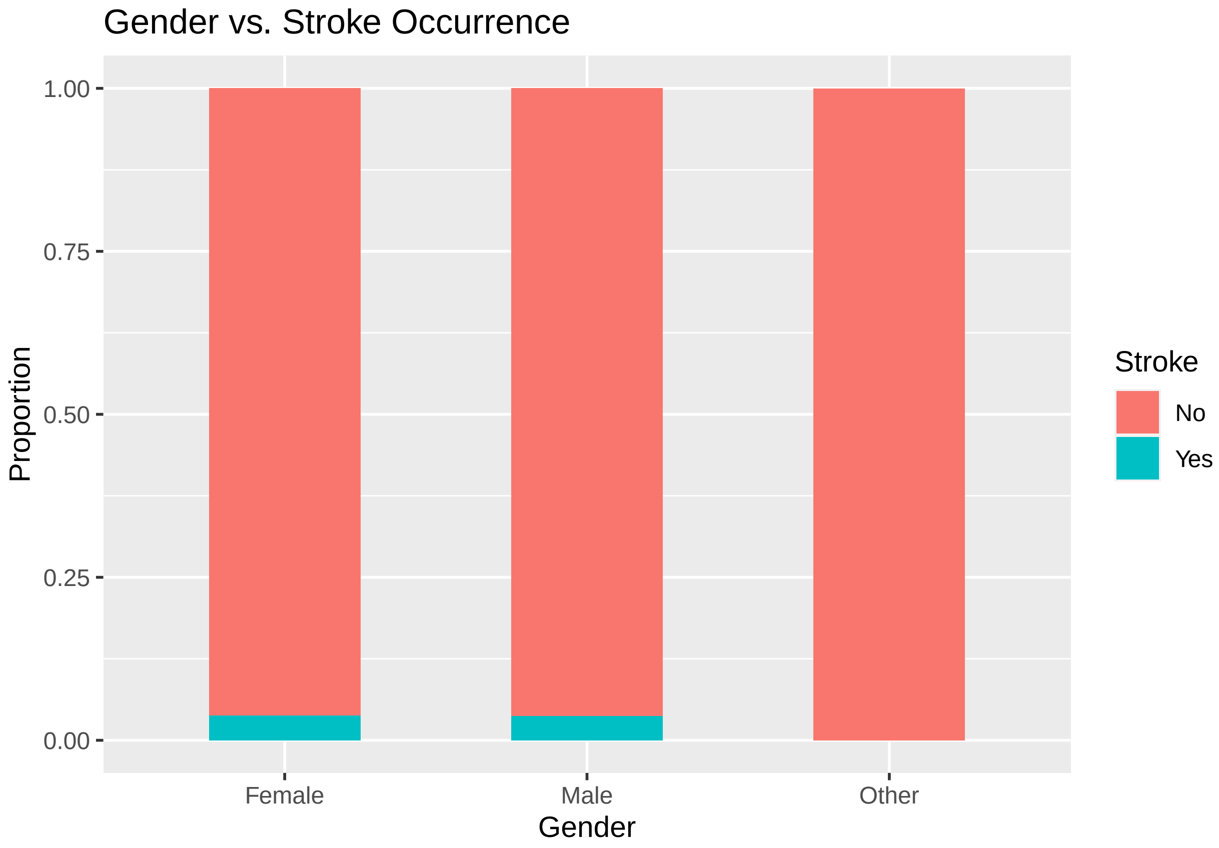
```
#Calculating correlations coefficients between features and stroke occurrence
correlation_coef<- cor(data[, c("age", "hypertension", "heart_disease",
                                "avg_glucose_level", "bmi")], data$stroke, use="complete.obs")
```

```
print(correlation_coef)
```

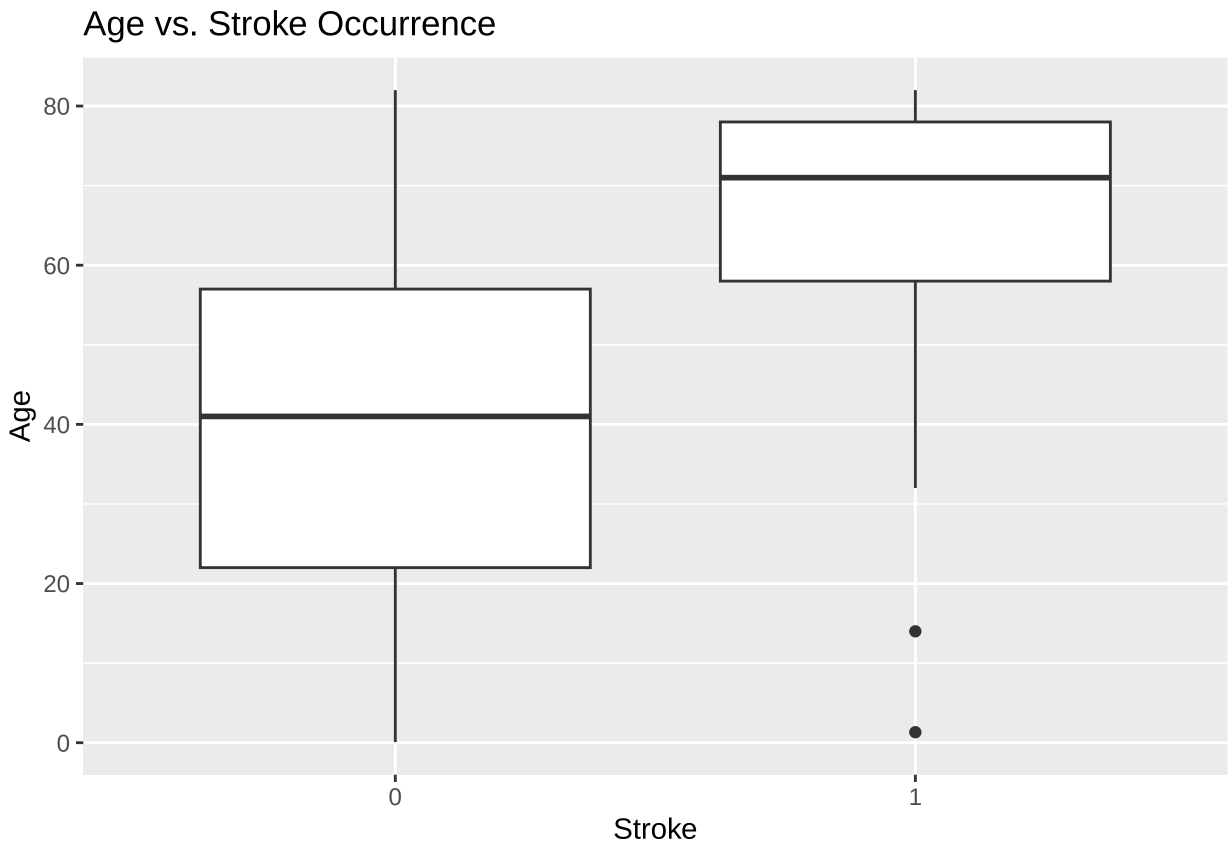
```
##                            [,1]
## age                 0.227160016
## hypertension        0.113353720
## heart_disease       0.089708947
## avg_glucose_level 0.003327288
## bmi                 0.034064605
```

From the correlation analysis coefficients, we observe that age ‘0.22’and hypertension ’0.11’ shows higher correlation with likelihood of stroke occurrence. Heart Disease ‘0.09’ has a small positive correlation with stroke risk, where as average glucose level ‘0.003’(negligible) and BMI ‘0.03’ have very low correlation with the likelihood of stroke occurrence.

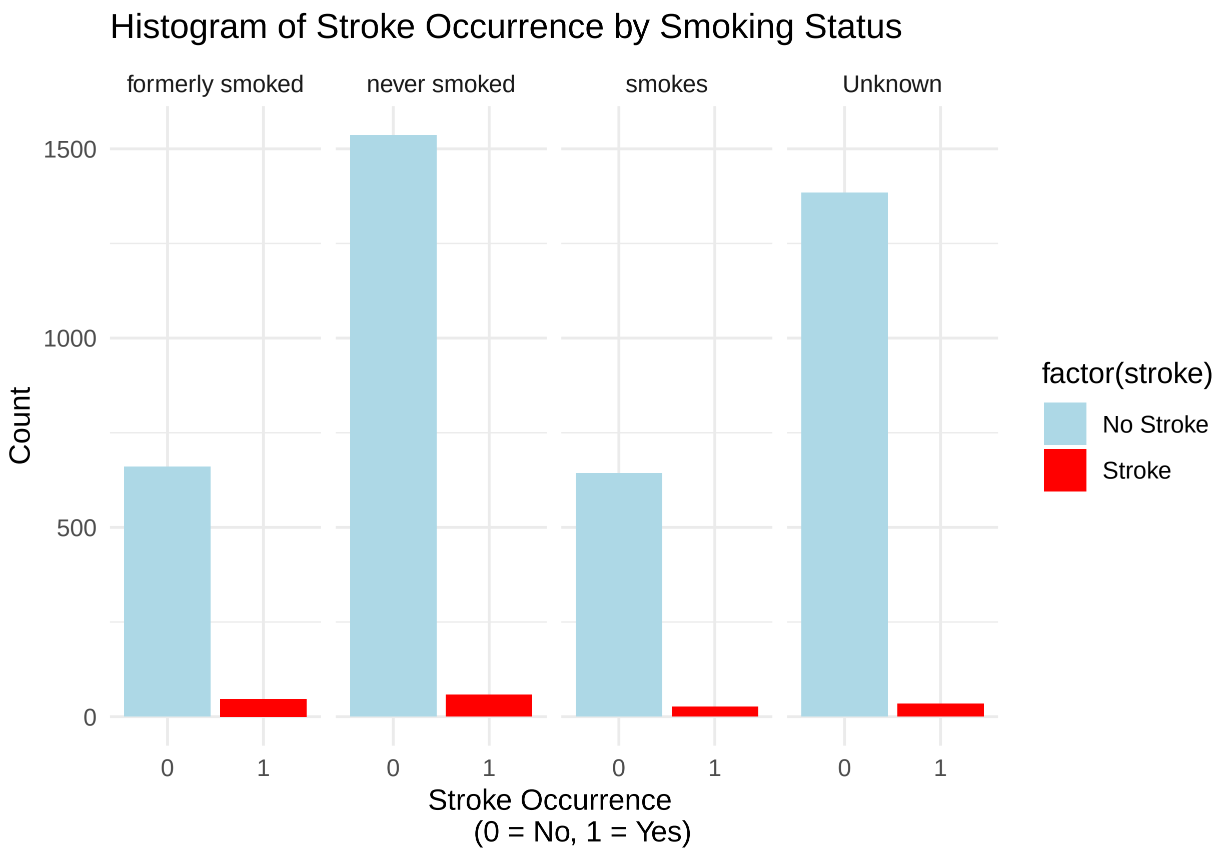
```
library(ggplot2)

#Bar plot for gender and stroke occurrence
ggplot(data, aes(x = gender, fill = as.factor(stroke))) +
  geom_bar(position = "fill", width = 0.5) +
  labs(title = "Gender vs. Stroke Occurrence", x = "Gender", y = "Proportion") +
  scale_fill_discrete(name = "Stroke", labels = c("No", "Yes"))
```



```r
#Box plot for age and stroke occurrence
ggplot(data, aes(x = as.factor(stroke), y = age)) +
  geom_boxplot() +
  labs(title = "Age vs. Stroke Occurrence", x = "Stroke", y = "Age")
```

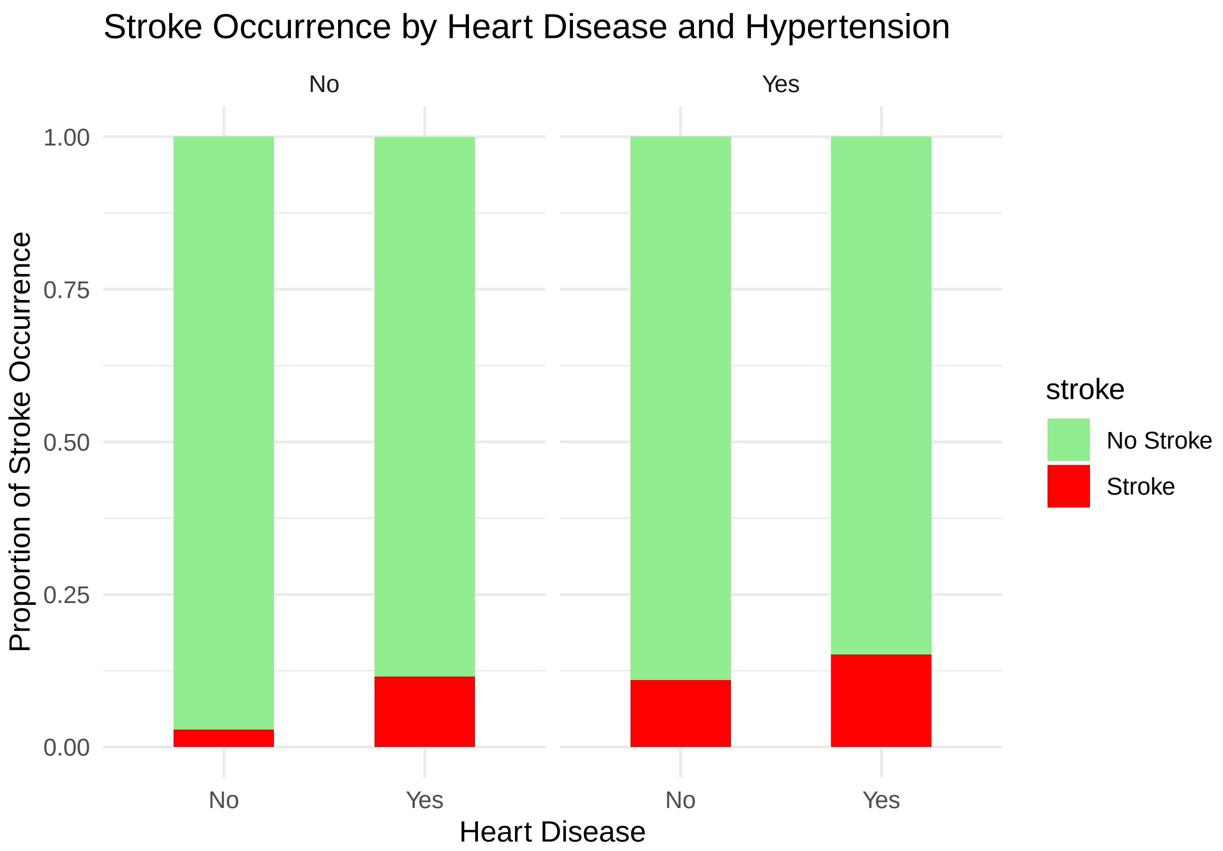


```r
#Histograms for Smoking Status and Stroke Occurrence
#Creating a subset of data for non-missing smoking status values
data_subset<- na.omit(data[data$smoking_status != "N/A", ])

smoking_stroke_hist_grid<- ggplot(data_subset, aes(x=factor(stroke), fill=factor(stroke)))+
  geom_bar(position = "dodge") +
  facet_grid(. ~ smoking_status, scales = "free_x") +
  labs(title = "Histogram of Stroke Occurrence by Smoking Status", x = "Stroke Occurrence
       (0 = No, 1 = Yes)", y = "Count") +
  scale_fill_manual(values=c("0"="lightblue", "1"="red"),labels=c("No Stroke", "Stroke")) +
  theme_minimal()

smoking_stroke_hist_grid
```

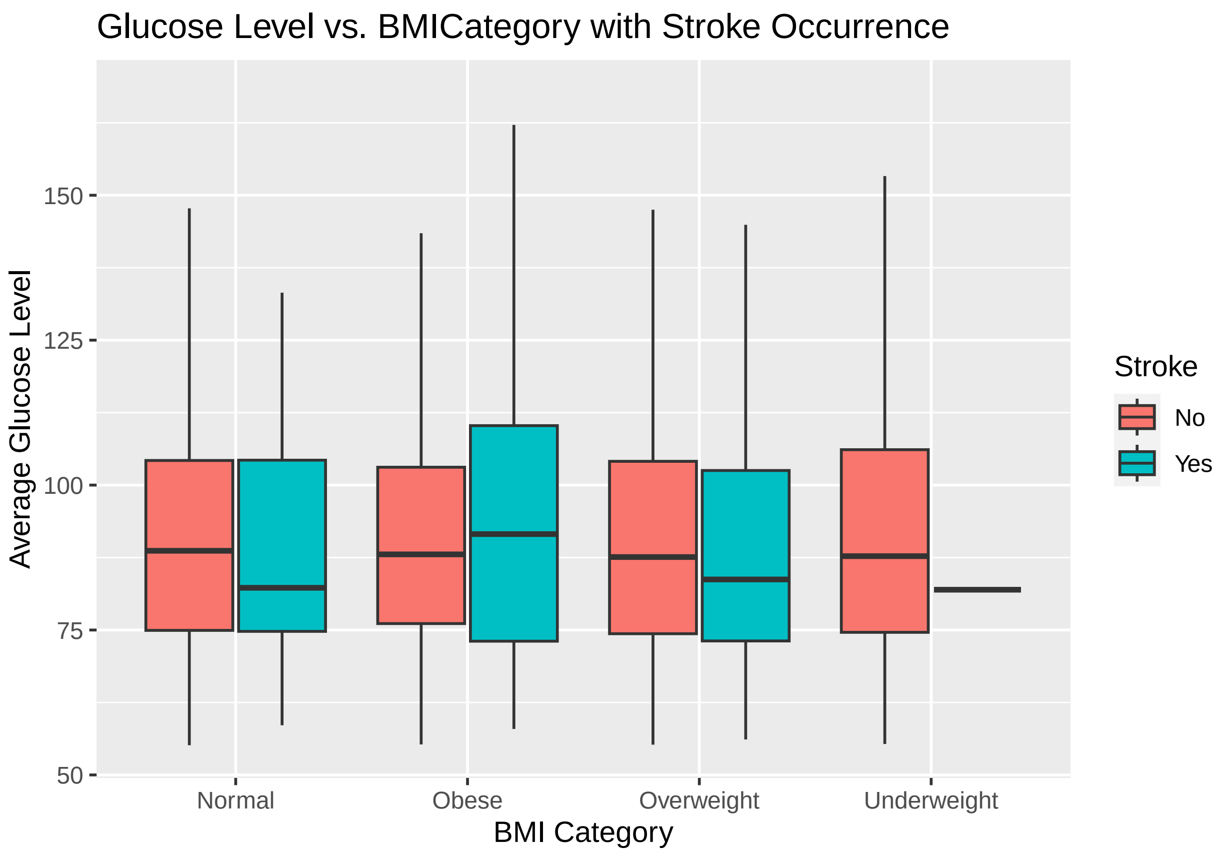


```
#Bar plot for Heart Disease, Hypertension, and Stroke
# Creating a subset of data for non-missing values in heart disease and hypertension
data_subset<- na.omit(data[data$heart_disease != "N/A" & data$hypertension != "N/A", ])

#Converting factors to appropriate types for plotting
data_subset$heart_disease <- factor(data_subset$heart_disease, levels = c("0", "1"),
                                    labels = c("No", "Yes"))
data_subset$hypertension <- factor(data_subset$hypertension, levels = c("0", "1"),
                                   labels = c("No", "Yes"))
data_subset$stroke <- factor(data_subset$stroke, levels = c("0", "1"),
                             labels = c("No Stroke", "Stroke"))

#Assuming 'hypertension' is a binary variable (0 or 1)
bar_plot <- ggplot(data_subset, aes(x = heart_disease, fill = stroke)) +
  geom_bar(position = "fill",width = 0.5) +
  facet_wrap(~hypertension) +
  labs(title = "Stroke Occurrence by Heart Disease and Hypertension",
       x = "Heart Disease", y = "Proportion of Stroke Occurrence") +
  scale_fill_manual(values = c("lightgreen", "red"),
                    labels = c("No Stroke", "Stroke")) +
  theme_minimal()
```

```
bar_plot
```



```
#Box plot for Glucose Level vs. BMICategory with Stroke Occurrence
ggplot(data, aes(x = BMICategory, y = avg_glucose_level, fill = as.factor(stroke))) +
  geom_boxplot(position = position_dodge(width = 0.8), outlier.shape = NA) +
  labs(title = "Glucose Level vs. BMICategory with Stroke Occurrence",
       x = "BMI Category", y = "Average Glucose Level", fill = "Stroke") +
  scale_fill_discrete(labels = c("No", "Yes"))
```



From the visual representation, we observe that the-

1) Bar plot for gender and stroke occurrence: This plot shows the proportion of stroke occurrence in each gender. The y-axis shows the gender, and the x-axis shows the proportion of stroke. The higher proportion of females have likelihood of stroke occurrence than the male. More than 80% of the male population did not has a stroke where as it is only likely to be 75% in female gender.

2) Box plot for age and stroke occurrence: This plot shows the age distribution to whether there is stroke or no stroke represented by 1 and 0. The x-axis shows the stroke, and the y-axis shows the age and the median is represented by line inside the box. People who had a stroke are older than people who have not had a stroke. The median age of people with stroke is 72, while the median age of people without stroke is 41. People with age 22-59 are likely to less chances to stroke risk where as the people with age 59-79 have higher likelihood to stroke occurrence.

3) Histograms for Smoking Status and Stroke Occurrence: This plot shows that the distribution of smoking status to stroke occurrence. The x-axis shows the smoking status in each category, and the y-axis shows the count. In ‘Formerly smoked’ category, only 20 people had a stroke and 210 people had no stroke. In ‘Never smoked’ category, only 30 people had a stroke and 720 people had no stroke. In ‘smokes’ category, only 10 people had a stroke and 305 people had no stroke. In ‘unknown’ category, only 15 people had a stroke and 660 people had no stroke.

4) Bar plot for Heart Disease, Hypertension, and Stroke: This plot shows that stroke risk by heart disease and hypertension represented 1 as 'yes' and 0 as 'no' on the top of the plot. It says that people with heart disease are more likely to have a stroke caused by hypertension than people without heart disease. In the category of No Hypertension, the likelihood of stroke risk is very low in No Heart disease compared to with heart disease. In Yes Hypertension category, people with heart disease are more likely to get a risk of stroke compared to no heart disease.

5) Box plot for Glucose Level vs. BMICategory with Stroke Occurrence: The plot shows the average glucose level vs. BMI category with stroke occurrence. The x-axis shows the BMICategory and y-axis shows the Glucose level. In 'Normal' BMI Category, the median of average glucose level without stroke is 73 and with stroke is 75. In 'Obese' BMI Category, the median of average glucose level without stroke is 76 and with stroke is 67. In 'Overweight' BMI Category, the median of average glucose level with stroke is 82 and with stroke is 80. In 'Underweight' BMI Category, the median of average glucose level with stroke is 78.

3) What metrics will be used to evaluate the quality of the data analysis?

1) Model prediction and Evaluation: Using train and testing sets, Logistic regression is performed on the model and is predicted. Calculating accuracy using by splitting the data into training and testing for evaluation.

2) Confusion Matrix: It is a table that presents a more detailed breakdown of correct and incorrect predictions, showing true positives, true negatives, false positives, and false negatives of the predicted and actual values.

3) Cross-Validation Scores: Evaluating the model's performance across multiple splits of the dataset through k-fold cross-validation, providing average values and their variance. It is validating the model's performance on unseen data to check for over fitting.

4) Uncertainty Estimation using Bootstrapping: Bootstrapping is a technique used for model validation and uncertainty estimation. On each resampled dataset, train regression models and construct prediction intervals or confidence intervals for model predictions. Analyze the variability and uncertainty in model performance measures throughout bootstrap iterations.

Data Analysis and Results [40 points]

1) Perform data analysis, document the analysis procedure, and evaluate the outcomes.
2) Present the data analysis results.
3) Interpret the results in a way to address the research questions.

Logistic Regression Model

```
#Building a logistic regression model
logit_model<- glm(stroke ~ age + hypertension + heart_disease + avg_glucose_level + bmi,
                  data = data, family = "binomial")

#Viewing model summary
summary(logit_model)
```

```
##
## Call:
## glm(formula = stroke ~ age + hypertension + heart_disease + avg_glucose_level +
##     bmi, family = "binomial", data = data)
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -7.339597   0.696495 -10.538   <2e-16 ***
## age                0.070861   0.005856  12.101   <2e-16 ***
## hypertension       0.493182   0.207535   2.376   0.0175 *
## heart_disease      0.109161   0.260762   0.419   0.6755
## avg_glucose_level  0.002759   0.003596   0.767   0.4428
## bmi               -0.005153   0.015619  -0.330   0.7415
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1406.3  on 4387  degrees of freedom
## Residual deviance: 1136.3  on 4382  degrees of freedom
## AIC: 1148.3
##
## Number of Fisher Scoring iterations: 7
```

Analysis Procedure: We are using Logistic Regression model as stroke occurrence is in binary form 1 or 0. This method helps in the estimation of the probability of stroke occurrence based on the given factors such as demographics, health and lifestyle and it is easier to interpret the impact of these factors on the likelihood of stroke risk. Stroke is a binary dependent variable & age, hypertension, heart_disease, avg_glucose_level and BMI are the independent variables. This model displays the summary of the fitted model. It consists of model coefficients, standard Errors, z value, p-values indicating if coefficients are significant and null deviance and residual deviance that are measures of model fit.

Result: From the output, we observe that age, hypertension are statistically significant with very low p-values. Their coefficients are positive meaning higher values are associated with higher stroke risk. Heart disease, glucose levels, BMI are not significant as their p-values>0.05. Also null deviance > Residual deviance which indicate a decent model fit. AIC provides a metric for model prediction. The factors- (i) Age coefficient of 0.064: It is statistically significant with the low p-value. (ii) Hypertension coefficient of 0.493: It is highly statistically significant with very low p-value. (iii) Heart disease, glucose and BMI are with positive and negative coefficients: These are not statistically significant.

Interpretation: Age and hypertension are most strongly correlated with higher stroke occurrences based on the highly significant positive coefficients in the model. Health factors such as Heart disease, glucose and BMI do not show significance.

Model Prediction and finding accuracy

```r
#predicting probabilities from the model
predicted_probabilities<- predict(logit_model, data, type = "response")

#Calculating accuracy for logistic regression
predicted_classes<- ifelse(predicted_probabilities > 0.5, 1, 0)
accuracy_logit<- mean(predicted_classes == data$stroke)
print(paste("Accuracy of Logistic Regression:", accuracy_logit))
```

```
## [1] "Accuracy of Logistic Regression: 0.962397447584321"
```

Analysis Procedure: The predict() function is used to get predicted probabilities from the fitted logistic model and predicted probabilities contains the probability of stroke occurrence for each sample which indicates that if probabilities > 0.5 then stroke occurrence is 1 and if probabilities <= 0.5 then stroke occurrence is 0. Accuracy is calculated by comparing the predicted classes to the actual stroke occurrences from original data.

Result: The output indicates that we are evaluating the accuracy of the logistic regression model which is 0.96. The accuracy value indicates the performed proportion of predictions from the model is correctly classifying the stroke occurrence of 96%, which demonstrates a very good performance of the model on likelihood of stroke risk. This logistic regression model is able to correctly predict the impact of demographic (age, gender)/health (hypertension, heart disease)/lifestyle (smoking status, BMI, average glucose level) factors on stroke occurrence.

Interpretation: The 96% accuracy prediction shows factors associated with stroke occurance. Age and hypertension diagnosis have the strongest correlations with increased strokes whereas, demographic health and lifestyle factors do not show significant correlation.

Checking accuracy by training and testing data

```r
#Building a logistic regression model on training data
logit_model_train<- glm(stroke ~ age + hypertension + heart_disease + avg_glucose_level + bmi,
                        data = train_data, family = "binomial")
```

```
#Predicting on training data
train_data$predicted_stroke_logit <- predict(logit_model, newdata = train_data, type = "response

#Calculating accuracy by training data
predicted_train <- ifelse(train_data$predicted_stroke_logit > 0.5, 1, 0)
accuracy_train <- mean(predicted_train == train_data$stroke)
print(paste("Training Accuracy:", accuracy_train))
```

```
## [1] "Training Accuracy: 0.964102564102564"
```

```
#Building a logistic regression model on test data
logit_model_test <- glm(stroke ~ age + hypertension + heart_disease + avg_glucose_level + bmi,
                        data = test_data, family = "binomial")

#Predicting on test data
test_data$predicted_stroke_logit <- predict(logit_model, newdata = test_data, type = "response")

#Calculating accuracy by test data
predicted_test <- ifelse(test_data$predicted_stroke_logit > 0.5, 1, 0)
accuracy_test <- mean(predicted_test == test_data$stroke)
print(paste("Testing Accuracy:", accuracy_test))
```

```
## [1] "Testing Accuracy: 0.955580865603645"
```

Procedure: This fits a logistic regression on the training data and testing to make predictions on train and test data, and calculates accuracy metrics. These accuracy of training and testing data is compared to the accuracy of the original data. The train and test data is obtained by splitting the original dataset into 80/20 split. Logistic model prediction and accuracy is done on both train and test data. The training accuracy is inflated, if we forecast on the same data that the model was fit on. Test accuracy provides an unbiased measure on previously unknown data. Comparing these with original accuracy for validation.

Result: From the output, we observe that the accuracy for training data is 0.96 and testing data is 0.95. The testing accuracy is very close to the training accuracy, there is only 0.9% difference. And these accuracy results are very similar to the original dataset accuracy. This shows very little over fitting.

Confusion matrix

```
#Confusion Matrix
conf_mat <- table(predicted = predicted_classes, actual = data$stroke)
print(conf_mat)
```

```
##          actual
## predicted    0    1
##         0 4223  165
```

Procedure: This forms a matrix with the predicted classes and stroke occurrence. A confusion matrix is a fundamental tool in evaluating the performance of a regression model where predictions are classified into different classes such as predicting stroke occurrence as 1 or 0. The breakdown prediction are- True Negative- Predicted no stroke, actually no stroke True Positive- Predicted stroke, actually stroke False Positive- Predicted stroke, actually no stroke False Negative- Predicted no stroke, actually stroke

Result: The output shows the predicted classes is 0 means no stroke predicted and 1 means stroke predicted The columns show actual classes is 0 means no stroke and 1 means stroke. The values 4223 is true negatives means No stroke correctly predicted 4223 times and 165 is false negatives means actual strokes incorrectly predicted as no stroke 165 times

Interpretation: The confusion matrix predicts no stroke for a large majority, 4223 of the no stroke cases which suggests the key variables of age and hypertension accurately identify most low risk cases. But, there are still 165 false negatives of actual strokes being missed.

Cross Validation for model evaluation

```
#Cross-validation with logistic regression
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.3.2
```

```
## Loading required package: lattice
```

```
#Defining control parameters
ctrl <- trainControl(method = "repeatedcv", repeats = 3)

#Converting 'stroke' to a factor with 2 levels
train_data$stroke <- factor(train_data$stroke, levels = c(0, 1))

#Training the model using cross-validation
cv_model <- train(stroke ~ age + hypertension + heart_disease + avg_glucose_level + bmi,
                  data = train_data, method = "glm", family = "binomial", trControl = ctrl)

summary(cv_model)
```

```
##
## Call:
## NULL
```

```
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -7.0192869  0.7771201  -9.032   <2e-16 ***
## age               0.0654035  0.0064353  10.163   <2e-16 ***
## hypertension      0.4962584  0.2359608   2.103   0.0355 *
## heart_disease     0.0693136  0.2949146   0.235   0.8142
## avg_glucose_level 0.0006947  0.0041714   0.167   0.8677
## bmi               0.0006141  0.0175518   0.035   0.9721
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1085.85  on 3509  degrees of freedom
## Residual deviance:  900.72  on 3504  degrees of freedom
## AIC: 912.72
##
## Number of Fisher Scoring iterations: 7
```

```
#View cross-validation results
print(cv_model)
```

```
## Generalized Linear Model
##
## 3510 samples
##    5 predictor
##    2 classes: ’0’, ’1’
##
## No pre-processing
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 3159, 3159, 3158, 3159, 3160, 3158, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.9641043  0
```

Procedure: Performing k-fold cross-validation can be very useful for assessing the robustness and generalizability of the regression model for evaluating the impact of various factors such as demographics, health and life style on stroke occurrence. The trainControl() sets up a resampling method called repeated k-fold CV. This repeated 3-fold CV is performed to evaluate a logistic regression in a more rigorous way by training and testing on different subsets of data. The output provides metrics on model generalization performance.

Result: We observe Generalized Linear Model which indicates logistic regression was used. The result gives average Cross- validation accuracy of 0.964. This is the Mean Cross-Validated Accuracy that is averaged from the 3 repeats which is very Close to original test accuracy and it is suggesting a minimal over fitting. Kappa statistic is 0.

Interpretation: The model stability on unseen folds shows age and hypertension have robust predictive relationships allowing accurate predictions. If other factors like heart disease, glucose had strong correlations, the model would over fit and performance would degrade substantially.

Bootstrapping

```
#Function to extract coefficients from the logistic regression model
coef_function <- function(data, indices) {
  fit <- glm(stroke ~ age + hypertension + heart_disease + avg_glucose_level + bmi,
             data = data[indices, ], family = "binomial")
  return(coef(fit))
}

#Performing bootstrapping for logistic regression
set.seed(123)  # Set seed for reproducibility
B <- 1000  # Number of bootstrap samples
bootstrap_results <- matrix(NA, nrow = B, ncol = 6)
#Columns to store bootstrap estimates for coefficients and intercept

for (i in 1:B) {
  sampled_data <- data[sample(nrow(data), replace = TRUE), ]

  #Extracting coefficients from bootstrapped logistic regression model
  bootstrap_coef <- coef_function(sampled_data, 1:nrow(sampled_data))
  bootstrap_results[i, ] <- bootstrap_coef
}

#Calculating the mean and confidence intervals of the bootstrap samples
bootstrap_estimates <- apply(bootstrap_results, 2, mean)
CI_low <- apply(bootstrap_results, 2, function(x) quantile(x, 0.025))
CI_high <- apply(bootstrap_results, 2, function(x) quantile(x, 0.975))

#Creating a data frame to display the results
output <- data.frame(Parameter = c("Intercept", "age", "hypertension",
                                    "heart_disease", "avg_glucose_level", "bmi"),
                     Estimate = bootstrap_estimates,
                     CI_low = CI_low,
                     CI_high = CI_high)

print(output)
```

```
##           Parameter     Estimate       CI_low      CI_high
## 1         Intercept -7.361259983 -8.723385184 -6.026760460
## 2               age  0.070835904  0.059242348  0.081865368
## 3      hypertension  0.499418693  0.064511548  0.886845706
## 4     heart_disease  0.096912595 -0.509332787  0.636040344
## 5 avg_glucose_level  0.002593433 -0.005489978  0.009825234
## 6               bmi -0.004351768 -0.034334649  0.025460555
```

Procedure: Uncertainty estimation in a regression model involves understanding the variability or confidence intervals around the estimated coefficients or predictions. This bootstrap sampling is used to estimate standard errors and confidence intervals for the logistic regression model coefficients. The mean of each coefficient across all samples is calculated to get bootstrap coefficient estimate. We are taking 2.5th and 97.5th percentile to get 95% CI bounds.

Result: From the result, we observe that the Intercept column Estimate column shows the original coefficient estimates, CI_low is the lower confidence interval and CI_high is the upper confidence interval. It is observed that Age and hypertension seem reliably significant showing impact while other factors are ambiguous without more data.The categories- (i) Age coefficient is 0.070 and 95% CI is (0.059, 0.081): The range indicates good precision and higher age increases stroke risk. (ii) Hypertension coefficient is 0.49 and CI is wider (0.064 to 0.88): The range indicates more variability in its estimates and coefficient shows hypertension diagnoses related with more strokes. (iii) Heart disease coefficient is 0.09 and CI is covering positive and negative (-0.50, 0.63): It indicates effect is ambiguous for heart disease and more data may be needed to precisely estimate its impact. (iv) Glucose and BMI are having CI's crossing 0: It indicates that they may not be truly significant.

Interpretation: The bootstrap analysis shows age and hypertension as having the most significant correlation to increased stroke likelihood based on the reliable positive effects.

Conclusion [15 points]

1) Does the analysis answer the research questions?

The various statistical analyses performance provides a comprehensive evidence that-

(i) Age and hypertension diagnosis shows most statistically significant positive correlation with increased odds of stroke occurrence based on the logistic regression coefficients and bootstrapping confidence intervals.

(ii) The ability to predict stroke occurrence with 96-97% accuracy using these demographic (age, gender)/health (hypertension) factors demonstrates substantial underlying correlations allowing such high performance even on unseen test data.

(iii) Other Health (heart disease)/lifestyle (smoking status, BMI, average glucose level)factors do not show significance and their inclusion does not meaningfully improve model generalization ability.

2) Discuss the scope and generalizability of the analysis.

Scope: The use of these techniques provides a comprehensive understanding of the logistic regression model's performance, predictive ability, stability, and uncertainty estimation.

(i) Logistic Regression: It is suitable for binary classification models, making it appropriate for predicting stroke occurrence (yes/no) based on demographic, health, and lifestyle factors.
(ii) Model prediction and accuracy: Training and testing the logistic regression model involve splitting the dataset into training and testing sets to evaluate its performance on unseen data.
(iii) Confusion matrix: It evaluates the model's classification performance by displaying true positives, true negatives, false positives, and false negatives.
(iv) Cross-Validation: It is the model's performance across multiple subsets of the dataset, ensuring it's robust and not over fitting to specific patterns.
(v) Bootstrapping: It estimates uncertainty around coefficients, providing confidence intervals for the impact of demographic, health, and lifestyle factors on stroke occurrence.

Generalizabiility: These analysis helps to build a robust, interpretable model for estimating stroke occurrence based on demographic, health, and lifestyle factors, enhancing the potential for generalization to new dataset.

(i) Logistic Regression: It interpret the relationships and provides coefficients indicating the impact of each factor on the likelihood of stroke occurrence.
(ii) Model prediction and accuracy: Finding accuracy on a test data provides an estimate of the model's performance on new tidy, unseen data. Higher accuracy suggests better generalizability, indicating that the model can predict stroke occurrence effectively.
(iii) Confusion matrix: It provides insights into the model's ability to correctly predict stroke occurrence and helps understand mis-classifications (e.g., false positives and false negatives).
(iv) Cross-Validation: Higher consistency in performance metrics across cross-validation folds indicates the model's stability and generalizability to different data subsets.
(v) Bootstrapping: Confidence intervals offer insights into the reliability and variability of estimated coefficients, enhancing the understanding of factors impact on stroke likelihood.

3) Discuss potential limitations and possibilities for improvement.

Limitations:

(i) Incomplete or biased data may lead to inaccuracies or biases in model outcomes.
(ii) The regression model might oversimplify the relationship between factors and stroke occurrence, potentially missing non-linear or interaction effects.
(iii) If the dataset has an imbalance in stroke occurrence, the model might favor the majority class and perform poorly in predicting the minority class.

Possibilities:

(i) Use of models other than logistic regression such as decision trees, random forests may capture more complex relationships.

(ii) Implementing techniques like oversampling, under sampling, or using evaluation metrics suited for imbalanced dataset could address class imbalance issues.

(iii) Performing external validation on different dataset or conducting sensitivity analysis could validate model performance and robustness.