**Methods Section for Logistic Regression Analysis**

**Data Acquisition**

We acquired the dataset: Stroke Prediction Dataset by Fedesorian in Kaggle. The dataset contained 1 outcome variable, stroke (1 if patient has stroke, 0 if not), and 10 predictors of stroke. The Predictors were (1) age, (2) gender (Male, Female, Other), (3) Hypertension (1 if the patient has hypertension, and 0 if not), (4) heart disease (1 if patient has heart disease, 0 if not) (5) patient ever married (yes or no), (6) work type (Government, private sector, self-employed, never worked), (7) residence type (rural or urban), (8) average glucose level (in the blood), (9) bmi (body mass index), (10) smoking status (never smoked, formerly smoked, smokes, and unknown. Each patient or row also has a unique id number. There were 5110 cases. The rule of thumb is 10-15 cases for each predictor. The predictors were either continuous such as age, and average glucose level, or categorical such as gender, heart disease, and marital status. Since the outcome variable had 2 values, and the predictors were of mixed type, We decided to use it to show Mgt and the research team how wide and complex logistic regression is.

**Data Cleaning**

Cleaning the data meant, (1) removing extraneous or irrelevant data, and (2) coding all variables as numeric. We first did a frequency distribution of all the variables to find out what all their values were. Once we found out (as shown above). We could move coding all variables as numeric, after removing any extraneous or irrelevant data. For example gender, we recoded other to N/A as we did smoking status for unknown.

Once we recoded all the extraneous values to N/A in all the variables, we deleted those cases and got our final dataset size of 3570.

We decided to code each predictor as the following:

(1) age (continuous), we decided to remake to numeric with 2 places after the decimal.

(2) gender (categorical) we coded 1 for male and 2 for female. There was only 1 case where it was coded other. We recoded other as N/A. We also recoded this predictor as numeric

(3) hypertension(categorical) was retyped to numeric

(4) heart disease(categorical) was retyped to numeric

(5) marital status (categorical) was recoded from yes to 1 and no to 2 and retyped as numeric,

(6) Work type(categorical) was recoded as 1 = Government , 2 = private sector, 3 = self-employed, 4 = never worked and then retyped as numeric

(7) residence type (categorical) was recoded as 1 = urban and 2 – rural. Then retyped as numeric.

(8) bmi (continuous) was retyped as numeric with 2 places after the decimal

(9) average glucose level(continuous) was retyped as numeric with 2 places after the decimal

(10) smoking status(categorical) was recoded as 1 = never smoked, 2 = formerly smoled and 3 = smokes, and unknown was recoded as N/A. After deletion of N/A the data was retyped as numeric.

(11) ID number -was left as is and deleted because its really not needed

(12) Stroke (outcome is categorical has 2 categories, 1 = stroke, 2 = no stroke

After we recoded all the variables to numbers and deleted all extraneous values. We retyped all variables as numeric with 2 places after the decimal.

**Exploratory Data Analysis**

We looked at a frequency distribution of the variables as well quartiles, mean median and mode thru the R commands of summary. In addition to look at all the individual values for each predictor, we saw that the dataset was unbalanced. There were few values of strokes coded 1 for stroke but a lot more 0’s. the dataset had 5.4% of the values coded for stroke. This did not reflect the real world of stroke in the US. Which was determined to be 3.1%. a 77% overreach for stroke in the model. So the dataset had stroke 1.82 times more than the actual US population. We could change the sampling, do SMOTE, etc., but a simpler fix is to change the weight for the intercept using the odds log formula. Once we determined the dataset reflected the real world for categorizing Stroke or no stroke we proceeded to develop the model. This change has real world implications of the MGT decided to finance a new stroke drug. We now can develop a plan around market share knowing the scope of the problem.

**Model Development**

The model we proposed to study was the logistic regression model.

**A screenshot of a computer

Description automatically generated**

**Odds Interpretation**

* Odds: p(x)1−p(x)1−*p*(*x*)*p*(*x*)
* Log Odds (Logit): log⁡(p(x)1−p(x))log(1−*p*(*x*)*p*(*x*))
* Exponentiating coefficients: Each βi*βi* represents the change in the log-odds for a one-unit increase in xi*xi*. eβi*eβi* is the odds ratio.

**Example**

Suppose you have predictors Age, Hypertension, SmokingStatus:

log⁡(p(x)1−p(x))=−8.42+0.03×Age+0.86×Hypertension+0.45×SmokingStatuslog(1−*p*(*x*)*p*

(*x*))=−8.42+0.03×Age+0.86×Hypertension+0.45×SmokingStatus

To get the stroke probability for a person:

z=−8.42+0.03×Age+0.86×Hypertension+0.45×SmokingStatusp(x)=11+e−z*z*=−8.42+0.03×Age+0.86×Hypertension+0.45×SmokingStatus*p*(*x*)=1+*e*−*z*1

**Model Evaluation and Validation**

Once we had the model developed. We needed to know how well it would fit the dataset. Since we are predicting the number of strokes, given the predictors we checked out and satisfied the assumptions of logistic regression.

• The outcome variable is binary (0 and 1)

• The observations were independent (each row represented a subject)

• The relationship between each continuous predictor and the log odds of the outcome should be linear

• No high multicollinearity**.**

**We then ran the model and reviewed its fit and predictive capabilities**

**(1) The model fit was done reviewed using the Hosmer\_lemesho statistic and Nagelkerke’s R**

**The Hosner lemesho found no predictor was P value < .05 so the fit is good.**

**Nagelkerke’s R and McFadden’s R had Pseudo R Squared at .18 and .21 which is**

**Within the rule of thumb for a logistic regression model**

**(2) The model’s predictive power was done by**

**(**1) calculating the AUC which was .8285 or the model could correctly discriminate between 0 and 1 83% of time.

(2) Calculating the Confusion Matrix for several threshold levels and calculating the sensitivity, specificity and F1

(3) Creating the ROC curve and finding the best point in tradeoff using Youden’s J to find the optimal balance between Sensitivity and Specificity.

**Tools and Environment**

**Results**