

Stroke Factors: Classification & Predictive Analytics

Danyang Liu (500936348). Supervisor: Dr. Ceni Babaoglu

3/7/2022

Dataset: removed NAs, removed outliers

Converted cat variables into num

No PCA analysis

Read dataset

```
stroke <- read.csv(file="stroke_1_raw.csv", header=T, sep=",")
```

Exploratory Analytics and Data Cleaning

```
# Descriptive analysis  
str(stroke)
```

```
## 'data.frame': 5110 obs. of 12 variables:  
## $ id : int 9046 51676 31112 60182 1665 56669 53882 10434 27419 60491 ...  
## $ gender : chr "Male" "Female" "Male" "Female" ...  
## $ age : num 67 61 80 49 79 81 74 69 59 78 ...  
## $ hypertension : int 0 0 0 0 1 0 1 0 0 0 ...  
## $ heart_disease : int 1 0 1 0 0 0 1 0 0 0 ...  
## $ ever_married : chr "Yes" "Yes" "Yes" "Yes" ...  
## $ work_type : chr "Private" "Self-employed" "Private" "Private" ...  
## $ Residence_type : chr "Urban" "Rural" "Rural" "Urban" ...  
## $ avg_glucose_level: num 229 202 106 171 174 ...  
## $ bmi : chr "36.6" "N/A" "32.5" "34.4" ...  
## $ smoking_status : chr "formerly smoked" "never smoked" "never smoked" "smokes" ...  
## $ stroke : int 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(stroke)
```

```
##      id      gender      age      hypertension  
## Min.   : 67  Length:5110      Min.   : 0.08  Min.   :0.00000  
## 1st Qu.:17741 Class :character  1st Qu.:25.00  1st Qu.:0.00000  
## Median :36932 Mode  :character Median :45.00  Median :0.00000  
## Mean   :36518                   Mean   :43.23  Mean   :0.09746  
## 3rd Qu.:54682                   3rd Qu.:61.00  3rd Qu.:0.00000
```

```

##   Max.    :72940                  Max.    :82.00  Max.    :1.00000
## heart_disease ever_married      work_type      Residence_type
## Min.    :0.00000 Length:5110      Length:5110  Length:5110
## 1st Qu.:0.00000 Class :character  Class :character  Class :character
## Median :0.00000 Mode  :character  Mode  :character  Mode  :character
## Mean    :0.05401
## 3rd Qu.:0.00000
## Max.    :1.00000
## avg_glucose_level   bmi          smoking_status   stroke
## Min.    : 55.12 Length:5110      Length:5110  Min.    :0.00000
## 1st Qu.: 77.25 Class :character  Class :character  1st Qu.:0.00000
## Median : 91.89 Mode  :character  Mode  :character  Median :0.00000
## Mean    :106.15
## 3rd Qu.:114.09
## Max.    :271.74

```

```

# Convert 'N/A's (strings) in dataset to NA
is.na(stroke) <- stroke == "N/A"
# Count number of NAs in dataset
sum(is.na(stroke))

```

```
## [1] 201
```

```

# Count number of NAs in all columns
colSums(is.na(stroke))

```

```

##           id      gender      age      hypertension
##           0        0        0        0
## heart_disease ever_married work_type Residence_type
##           0        0        0        0
## avg_glucose_level   bmi      smoking_status   stroke
##           0        201        0        0

```

```

# Count number of 'Unknown's in all columns
colSums(stroke == "Unknown")

```

```

##           id      gender      age      hypertension
##           0        0        0        0
## heart_disease ever_married work_type Residence_type
##           0        0        0        0
## avg_glucose_level   bmi      smoking_status   stroke
##           0        NA       1544        0

```

```

# Remove first column 'id'; irrelevant to data analysis
stroke <- stroke[2:12]

```

```

# Check attribute levels and convert data types to numeric
# For binary "Yes"/"No" values, "Yes" = 1 and "No" = 2
str(stroke)

```

```
## 'data.frame': 5110 obs. of 11 variables:
```

```

## $ gender      : chr "Male" "Female" "Male" "Female" ...
## $ age         : num 67 61 80 49 79 81 74 69 59 78 ...
## $ hypertension : int 0 0 0 0 1 0 1 0 0 0 ...
## $ heart_disease : int 1 0 1 0 0 0 1 0 0 0 ...
## $ ever_married : chr "Yes" "Yes" "Yes" "Yes" ...
## $ work_type    : chr "Private" "Self-employed" "Private" "Private" ...
## $ Residence_type : chr "Urban" "Rural" "Rural" "Urban" ...
## $ avg_glucose_level: num 229 202 106 171 174 ...
## $ bmi          : chr "36.6" NA "32.5" "34.4" ...
## $ smoking_status : chr "formerly smoked" "never smoked" "never smoked" "smokes" ...
## $ stroke       : int 1 1 1 1 1 1 1 1 1 1 ...

unique(stroke$gender)

## [1] "Male"   "Female" "Other"

stroke$gender <- gsub("Male", 1, stroke$gender)
stroke$gender <- gsub("Female", 2, stroke$gender)
stroke$gender <- gsub("Other", 3, stroke$gender)
stroke$gender <- as.numeric(stroke$gender)
unique(stroke$gender)

## [1] 1 2 3

unique(stroke$ever_married)

## [1] "Yes" "No"

stroke$ever_married <- gsub("Yes", 1, stroke$ever_married)
stroke$ever_married <- gsub("No", 0, stroke$ever_married)
stroke$ever_married <- as.numeric(stroke$ever_married)
unique(stroke$ever_married)

## [1] 1 0

unique(stroke$work_type)

## [1] "Private"      "Self-employed" "Govt_job"      "children"
## [5] "Never_worked"

stroke$work_type <- gsub("Private", 1, stroke$work_type)
stroke$work_type <- gsub("Self-employed", 2, stroke$work_type)
stroke$work_type <- gsub("Govt_job", 3, stroke$work_type)
stroke$work_type <- gsub("children", 4, stroke$work_type)
stroke$work_type <- gsub("Never_worked", 5, stroke$work_type)
stroke$work_type <- as.numeric(stroke$work_type)
unique(stroke$work_type)

## [1] 1 2 3 4 5

```

```

unique(stroke$Residence_type)

## [1] "Urban" "Rural"

stroke$Residence_type <- gsub("Urban", 1, stroke$Residence_type)
stroke$Residence_type <- gsub("Rural", 2, stroke$Residence_type)
stroke$Residence_type <- as.numeric(stroke$Residence_type)
unique(stroke$Residence_type)

## [1] 1 2

stroke$bmi <- as.numeric(stroke$bmi)

unique(stroke$smoking_status)

## [1] "formerly smoked" "never smoked"      "smokes"           "Unknown"

stroke$smoking_status <- gsub("formerly smoked", 1, stroke$smoking_status)
stroke$smoking_status <- gsub("never smoked", 2, stroke$smoking_status)
stroke$smoking_status <- gsub("smokes", 3, stroke$smoking_status)
stroke$smoking_status <- gsub("Unknown", 4, stroke$smoking_status)
stroke$smoking_status <- as.numeric(stroke$smoking_status)
unique(stroke$smoking_status)

## [1] 1 2 3 4

# Assign "No Stroke" and "Stroke" labels for Stroke attribute
stroke$stroke <- ifelse(stroke$stroke == 0, "No Stroke", "Stroke")
# Assign Stroke values as factor levels
stroke$stroke <- as.factor(stroke$stroke)

# Check that all attributes are now numeric data types
str(stroke)

## 'data.frame': 5110 obs. of 11 variables:
##   $ gender       : num  1 2 1 2 2 1 1 2 2 2 ...
##   $ age          : num  67 61 80 49 79 81 74 69 59 78 ...
##   $ hypertension : int  0 0 0 0 1 0 1 0 0 0 ...
##   $ heart_disease: int  1 0 1 0 0 0 1 0 0 0 ...
##   $ ever_married : num  1 1 1 1 1 1 0 1 1 ...
##   $ work_type    : num  1 2 1 1 2 1 1 1 1 1 ...
##   $ Residence_type: num  1 2 2 1 2 1 2 1 2 1 ...
##   $ avg_glucose_level: num  229 202 106 171 174 ...
##   $ bmi          : num  36.6 NA 32.5 34.4 24 29 27.4 22.8 NA 24.2 ...
##   $ smoking_status: num  1 2 2 3 2 1 2 2 4 4 ...
##   $ stroke       : Factor w/ 2 levels "No Stroke","Stroke": 2 2 2 2 2 2 2 2 2 2 ...

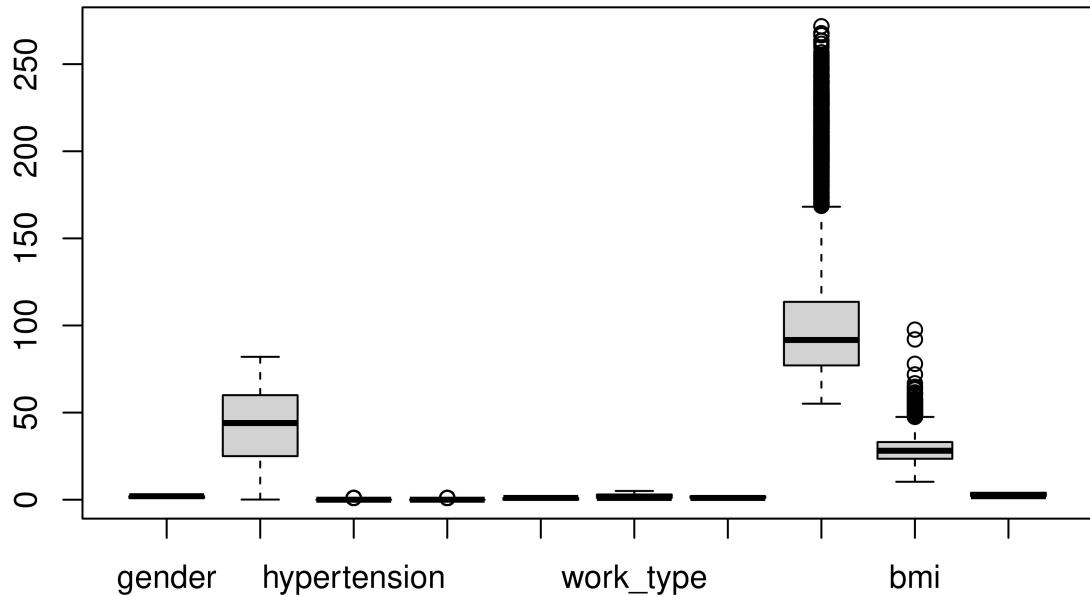
```

```

# Deal with NAs
# Method 1: remove NAs
stroke_noNAs <- stroke[complete.cases(stroke), ]
# Method 2: replace NAs with values using k-NN algorithm?

# Deal with outliers
# Box plot to visualize outliers
boxplot(as.matrix(stroke_noNAs[1:10]))

```



```

# Excluding categorical variables, avg_glucose_level
# And bmi have several outliers
# Remove outliers using interquartile range values
agl_outliers <- boxplot(stroke$avg_glucose_level, plot = FALSE)$out
bmi_outliers <- boxplot(stroke$bmi, plot = FALSE)$out
stroke_noNAs_noOL <- stroke_noNAs
stroke_noNAs_noOL <- stroke_noNAs_noOL[-which(stroke_noNAs_noOL$avg_glucose_level %in% agl_outliers),]
stroke_noNAs_noOL <- stroke_noNAs_noOL[-which(stroke_noNAs_noOL$bmi %in% bmi_outliers),]

# Examine correlations between all Independent Variables
cor(stroke_noNAs_noOL[1:10])

```

	gender	age	hypertension	heart_disease
## gender	1.0000000000	0.047163661	-0.0181116010	-0.087077746
## age	0.0471636606	1.0000000000	0.2492046322	0.236193434
## hypertension	-0.0181116010	0.249204632	1.0000000000	0.106065206

```

## heart_disease      -0.0870777462  0.236193434  0.1060652062  1.0000000000
## ever_married       0.0508315382  0.687498881  0.1488340141  0.105364898
## work_type          -0.0758616230 -0.439614390 -0.0721676438 -0.041084225
## Residence_type     -0.0003523739 -0.009598891  0.0038834139  0.014064422
## avg_glucose_level -0.0305248091 -0.023924488 -0.0009078475  0.004947325
## bmi                 0.0054726191  0.378683833  0.1515384482  0.054618944
## smoking_status     -0.0590370218 -0.385509590 -0.1155068859 -0.057584935
##                           ever_married   work_type Residence_type avg_glucose_level
## gender              0.0508315382 -0.07586162  -0.0003523739  -0.0305248091
## age                 0.6874988811 -0.43961439  -0.0095988913  -0.0239244877
## hypertension         0.1488340141 -0.07216764  0.0038834139  -0.0009078475
## heart_disease        0.1053648985 -0.04108422  0.0140644220  0.0049473252
## ever_married         1.0000000000 -0.39116104  0.0004186879  -0.0083602287
## work_type            -0.3911610411  1.00000000  -0.0155902656  0.0109823333
## Residence_type       0.0004186879 -0.01559027  1.0000000000  0.0145557947
## avg_glucose_level   -0.0083602287  0.01098233  0.0145557947  1.0000000000
## bmi                 0.3756328526 -0.38386175  -0.0110487374  0.0017839920
## smoking_status      -0.3177225122  0.33765019  -0.0042874498  0.0178261247
##                           bmi   smoking_status
## gender              0.005472619  -0.05903702
## age                 0.378683833  -0.38550959
## hypertension         0.151538448  -0.11550689
## heart_disease        0.054618944  -0.05758493
## ever_married         0.375632853  -0.31772251
## work_type            -0.383861746  0.33765019
## Residence_type       -0.011048737  -0.00428745
## avg_glucose_level   0.001783992  0.01782612
## bmi                 1.0000000000  -0.26338455
## smoking_status      -0.263384550  1.00000000
```

```

# Normalize continuous numeric variables
# Such as age, avg_blood_glucose, and bmi
# Using z-score methods
stroke_noNAs_noOL$age <- (stroke_noNAs_noOL$age - mean(stroke_noNAs_noOL$age))/sd(stroke_noNAs_noOL$age)
stroke_noNAs_noOL$avg_glucose_level <- (stroke_noNAs_noOL$avg_glucose_level - mean(stroke_noNAs_noOL$avg_glucose_level))/sd(stroke_noNAs_noOL$avg_glucose_level)
stroke_noNAs_noOL$bmi <- (stroke_noNAs_noOL$bmi - mean(stroke_noNAs_noOL$bmi))/sd(stroke_noNAs_noOL$bmi)
```

Classification

Predictive Analytics: Logistic Regression

```

# Split dataset into 70% training, 30% testing sets
stroke_index1 <- sample(1:nrow(stroke_noNAs_noOL), 0.7 * nrow(stroke_noNAs_noOL))

# Assign selected sample as training set
# Assign leftover dataset as test set
train.set1 <- stroke_noNAs_noOL[stroke_index1,]
test.set1 <- stroke_noNAs_noOL[-stroke_index1,]

# Logistic regression model for prediction
glm_model1 <- glm(formula = stroke~., data = train.set1, family = "binomial")
summary(glm_model1)
```

```

## 
## Call:
## glm(formula = stroke ~ ., family = "binomial", data = train.set1)
## 
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.9005  -0.2596  -0.1512  -0.0885   3.5382 
## 
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)    
## (Intercept)           -4.057543  0.680977 -5.958 2.55e-09 *** 
## gender                -0.108863  0.226937 -0.480  0.63144    
## age                   1.345140  0.166815  8.064 7.40e-16 *** 
## hypertension          0.783931  0.278063  2.819  0.00481 **  
## heart_disease         0.269928  0.364550  0.740  0.45903    
## ever_married          0.008993  0.338160  0.027  0.97878    
## work_type              -0.078070 0.148471 -0.526  0.59901    
## Residence_type         0.122204  0.220478  0.554  0.57939    
## avg_glucose_level     -0.020754 0.109962 -0.189  0.85030    
## bmi                   -0.162565  0.139428 -1.166  0.24364    
## smoking_status        -0.069109  0.107976 -0.640  0.52215    
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
## Null deviance: 821.19  on 2981  degrees of freedom 
## Residual deviance: 683.22  on 2971  degrees of freedom 
## AIC: 705.22 
## 
## Number of Fisher Scoring iterations: 8

```

Evaluation Metrics

```

predicted1 <- predict(glm_model1, test.set1, type = "response")
# Setting 0.5 as threshold - binary prediction
predicted_class1 <- ifelse(predicted1 >= 0.5, "Stroke", "No Stroke")
ConfusionMatrix1 <- table(actual = test.set1$stroke, predicted = predicted_class1)
ConfusionMatrix1

##           predicted
## actual      No Stroke
##   No Stroke      1235
##   Stroke          44

```

Abysmal predictions using only logistic regression applied to dataset with NAs removed (from BMI column) and outliers removed. No strokes are predicted at all