Part-2

Florencia Luque and Seyed Amirhossein Mosaddad

2024-10-18

Part 2

```
library(caret)
library(caretEnsemble)
library(h2o)
library(tidymodels)
library(ROCR)
library(ConfusionTableR)
library(dplyr)
```

```
path = "/Users/soroush/Desktop/UC3M/courses/R/minitask1/Stroke_data"
stroke_data = read.csv("healthcare-dataset-stroke-data.csv",header = TRUE)

stroke_data = na.omit(stroke_data)
stroke_data = stroke_data[stroke_data$gender!="Other",]
stroke_data$gender = as.factor(stroke_data$gender)
stroke_data$hypertension = as.factor(stroke_data$hypertension)
stroke_data$heart_disease = as.factor(stroke_data$heart_disease)
stroke_data$work_type = as.factor(stroke_data$work_type)
stroke_data$Residence_type = as.factor(stroke_data$Residence_type)
stroke_data$ever_married = as.factor(stroke_data$ever_married)
stroke_data$smoking_status = as.factor(stroke_data$smoking_status)
stroke_data$stroke = as.factor(stroke_data$stroke)
stroke_data$bmi =as.numeric(stroke_data$bmi)

frec_table <- cut(stroke_data$bmi, breaks = c(0, 18.5, 24.9, 29.9, 34.9, Inf),
                labels = c("Underweight", "Normal", "Overweight", "Obese", "Extremely Obese"))
stroke_data$cat_weight = frec_table

glucose_categories <- cut(stroke_data$avg_glucose_level,
                          breaks = c(0, 99.9, 125.9, Inf),
                          labels = c("Normal", "Prediabetes", "Diabetes"))

# Reorder the levels to put Normal in the middle
glucose_categories <- factor(glucose_categories, levels = c("Prediabetes", "Normal", "Diabetes"))

# Add the categories to the data frame
stroke_data$glucose_category = glucose_categories
str(stroke_data)
```

```
## 'data.frame':    5109 obs. of  14 variables:
##  $ id                : int  9046 51676 31112 60182 1665 56669 53882 10434 27419 60491 ...
##  $ gender            : Factor w/ 2 levels "Female","Male": 2 1 2 1 1 2 2 1 1 1 ...
##  $ age               : num  67 61 80 49 79 81 74 69 59 78 ...
##  $ hypertension      : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 2 1 1 1 ...
##  $ heart_disease     : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 2 1 1 1 ...
##  $ ever_married      : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 2 2 1 2 2 ...
##  $ work_type         : Factor w/ 5 levels "children","Govt_job",..: 4 5 4 4 5 4 4 4 4 4 ...
##  $ Residence_type    : Factor w/ 2 levels "Rural","Urban": 2 1 1 2 1 2 1 2 1 2 ...
##  $ 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    : Factor w/ 4 levels "formerly smoked",..: 1 2 2 3 2 1 2 2 4 4 ...
##  $ stroke            : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
##  $ cat_weight        : Factor w/ 5 levels "Underweight",..: 5 NA 4 4 2 3 3 2 NA 2 ...
##  $ glucose_category : Factor w/ 3 levels "Prediabetes",..: 3 3 1 3 3 3 2 2 2 2 ...
```

Relation between variables

```
tab_st_gd = table(stroke_data$stroke,stroke_data$gender)
chisq.test(tab_st_gd)
```

Stroke with Gender

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_st_gd
## X-squared = 0.34, df = 1, p-value = 0.5598
```

The *p-value* is larger than 0.05 this mean that there's not evidence of dependency between the variables gender and stroke.

```
tab_st_hy = table(stroke_data$stroke,stroke_data$hypertension)
chisq.test(tab_st_hy)
```

Stroke with Hypertension

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_st_hy
## X-squared = 81.573, df = 1, p-value < 2.2e-16
```

The *p-value* is a lot smaller than 0.05. This mean that there's a relation between getting a stroke and hypertension. This is a could be a comprobation of the hypothesis that we established earlier about the existence of a relation between this two variables.

```r
tab_st_hd = table(stroke_data$stroke,stroke_data$heart_disease)
chisq.test(tab_st_hd)
```

Stroke with Heart Disease

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_st_hd
## X-squared = 90.229, df = 1, p-value < 2.2e-16
```

The *p-value* is a lot smaller than 0.05. This mean that there's a relation between getting a stroke and heart disease This is a could be a comprobation of the hypothesis that we established earlier about the existence of a relation between this two variables.

```r
tab_st_rt = table(stroke_data$stroke,stroke_data$Residence_type)
chisq.test(tab_st_rt)
```

Stroke with Residence Type

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_st_rt
## X-squared = 1.075, df = 1, p-value = 0.2998
```

As we had seen in the graph there´s no evidence to say that there's a relation between the type of residence and getting a stroke.

```r
tab_st_em = table(stroke_data$stroke,stroke_data$ever_married)
chisq.test(tab_st_em)
```

Stroke with Ever Married

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_st_em
## X-squared = 58.868, df = 1, p-value = 1.686e-14
```

Apparently there's a relation within this two variables. Having a stroke have a relation with have been or had been ever married.

```r
tab_st_sk = table(stroke_data$stroke,stroke_data$smoking_status)
chisq.test(tab_st_sk)
```

Stroke with Smoking Status

```
##
##  Pearson's Chi-squared test
##
## data:  tab_st_sk
## X-squared = 29.226, df = 3, p-value = 2.008e-06
```

The *p-value* is a lot smaller than 0.05 so there's is a relation between the variables. This was something that we *dont know what to write here*

```r
tab_st_wt = table(stroke_data$stroke,stroke_data$work_type)
chisq.test(tab_st_wt)
```

Stroke with Work Type

```
##
##  Pearson's Chi-squared test
##
## data:  tab_st_wt
## X-squared = 49.159, df = 4, p-value = 5.409e-10
```

There's a relation between the variables ($p\text{-}value<0.05$). This we think was because of the difference between the quantity of people who got a stroke and work independently and the people who work with children because the difference was big between them.

```r
tab_hy_hd = table(stroke_data$heart_disease,stroke_data$hypertension)
chisq.test(tab_hy_hd)
```

Heart Disease and Hypertension

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_hy_hd
## X-squared = 58.31, df = 1, p-value = 2.239e-14
```

There's a relation between hypertension and heart disease and both variable are related to stroke. This could be a good indicator that within only one of this variables we could have the same information in the model.

```r
tab_hd_em = table(stroke_data$heart_disease,stroke_data$ever_married)
chisq.test(tab_hd_em)
```

Heart Disease and Ever Married

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_hd_em
## X-squared = 66.036, df = 1, p-value = 4.428e-16
```

```r
tab_hd_sk= table(stroke_data$heart_disease,stroke_data$smoking_status)
chisq.test(tab_hd_sk)
```

Heart Disease and Smoking Status

```
##
##  Pearson's Chi-squared test
##
## data:  tab_hd_sk
## X-squared = 44.74, df = 3, p-value = 1.051e-09
```

```r
tab_hd_wt = table(stroke_data$heart_disease,stroke_data$work_type)
chisq.test(tab_hd_wt)
```

Heart Disease and Work Type

```
##
##  Pearson's Chi-squared test
##
## data:  tab_hd_wt
## X-squared = 70.689, df = 4, p-value = 1.623e-14
```

```r
tab_hy_em = table(stroke_data$heart_disease,stroke_data$ever_married)
chisq.test(tab_hy_em)
```

Hypertension and Ever Married

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab_hy_em
## X-squared = 66.036, df = 1, p-value = 4.428e-16
```

```r
tab_hy_sk = table(stroke_data$heart_disease,stroke_data$smoking_status)
chisq.test(tab_hy_sk)
```

Hypertension and Smoking Status

```
##
##  Pearson's Chi-squared test
##
## data:  tab_hy_sk
## X-squared = 44.74, df = 3, p-value = 1.051e-09
```

```r
tab_hy_wt= table(stroke_data$heart_disease,stroke_data$work_type)
chisq.test(tab_hy_wt)
```

Hypertension and Work Type

```
##
##  Pearson's Chi-squared test
##
## data:  tab_hy_wt
## X-squared = 70.689, df = 4, p-value = 1.623e-14
```

Stroke with Age

```r
t.test(age ~ stroke, data = stroke_data)
```

```
##
##  Welch Two Sample t-test
##
## data:  age by stroke
## t = -29.682, df = 331.68, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -27.46015 -24.04658
## sample estimates:
## mean in group 0 mean in group 1
##        41.97483        67.72819
```

As you can see we reject the null hypothesis that said that both group have the same mean and this tell as that this variable could have and impact in the probability of getting a stroke in this case getting older increase your chances.

Stroke with Average Glucose Levels

```r
t.test(avg_glucose_level ~ stroke, data = stroke_data)
```

```
##
##  Welch Two Sample t-test
##
## data:  avg_glucose_level by stroke
## t = -6.9844, df = 260.9, p-value = 2.373e-11
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -35.58269 -19.93162
## sample estimates:
## mean in group 0 mean in group 1
##        104.7876        132.5447
```

Also we can say that the difference in means between the groups with a stroke are without is not zero.

Stroke with BMI

```r
t.test(bmi ~ stroke, data = stroke_data)
```

```
##
##  Welch Two Sample t-test
##
## data:  bmi by stroke
## t = -3.6374, df = 237.84, p-value = 0.0003377
## alternative hypothesis: true difference in means between group 0 and group 1 is not equal to 0
## 95 percent confidence interval:
##  -2.5387991 -0.7549231
## sample estimates:
## mean in group 0 mean in group 1
##        28.82443        30.47129
```

Apparently all the continuous variables could have and impact in the chances or getting a stroke. This variable *bmi* also have a *p-value* less than 0.05 so we can say that the groups have significant different means.

Models

Logistic Regresion (Flo)

```r
set.seed(23)
index_split = createDataPartition(stroke_data$stroke,p=0.8,list=FALSE)
train = stroke_data[index_split,]
test = stroke_data[-index_split,]
train = subset(train,select = -c(id,cat_weight,glucose_category))
train = na.omit(train)
test = subset(test,select = -c(id,cat_weight,glucose_category))
test = na.omit(test)
```

We would start the models with one that includes all the variables and the reduce it with different test.

```
class_weight <- ifelse(train$stroke == 1, 25, 1.04)
log_reg_model = glm(stroke~(gender + age + hypertension + heart_disease + ever_married +
    work_type + Residence_type + avg_glucose_level + bmi + smoking_status),data = train,family = binomi
summary(log_reg_model)
```

```
##
## Call:
## glm(formula = stroke ~ (gender + age + hypertension + heart_disease +
##     ever_married + work_type + Residence_type + avg_glucose_level +
##     bmi + smoking_status), family = binomial(link = "logit"),
##     data = train, weights = class_weight)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -3.860e+00  2.485e-01 -15.532  < 2e-16 ***
## genderMale                  -5.040e-02  6.005e-02  -0.839 0.401304
## age                          7.914e-02  2.321e-03  34.104  < 2e-16 ***
## hypertension1                8.490e-01  7.759e-02  10.942  < 2e-16 ***
## heart_disease1               3.797e-01  1.003e-01   3.784 0.000154 ***
## ever_marriedYes             -1.144e-01  9.289e-02  -1.231 0.218197
## work_typeGovt_job           -1.359e+00  2.611e-01  -5.204 1.95e-07 ***
## work_typeNever_worked       -1.123e+01  1.329e+02  -0.084 0.932663
## work_typePrivate            -1.292e+00  2.549e-01  -5.070 3.98e-07 ***
## work_typeSelf-employed      -1.552e+00  2.679e-01  -5.793 6.91e-09 ***
## Residence_typeUrban          3.577e-02  5.762e-02   0.621 0.534725
## avg_glucose_level            3.118e-03  5.853e-04   5.326 1.00e-07 ***
## bmi                          1.599e-02  4.514e-03   3.543 0.000395 ***
## smoking_statusnever smoked -3.382e-01  7.671e-02  -4.409 1.04e-05 ***
## smoking_statussmokes         2.248e-01  8.857e-02   2.538 0.011138 *
## smoking_statusUnknown       -5.600e-01  9.272e-02  -6.040 1.54e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11203.1  on 3928  degrees of freedom
## Residual deviance:  7500.3  on 3913  degrees of freedom
## AIC: 7377.4
##
## Number of Fisher Scoring iterations: 12
```

```
fitted.results <- predict(log_reg_model,newdata = test,type='response')
fitted.results <- ifelse(fitted.results > 0.8,1,0)
test$accu = fitted.results
misClasificError <- mean(fitted.results != test$stroke)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.892747701736466"
```

```r
confusionMatrix(as.factor(test$accu),test$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 851  19
##          1  86  23
##
##                Accuracy : 0.8927
##                  95% CI : (0.8717, 0.9114)
##     No Information Rate : 0.9571
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2587
##
##  Mcnemar's Test P-Value : 1.187e-10
##
##             Sensitivity : 0.9082
##             Specificity : 0.5476
##          Pos Pred Value : 0.9782
##          Neg Pred Value : 0.2110
##              Prevalence : 0.9571
##          Detection Rate : 0.8693
##    Detection Prevalence : 0.8887
##       Balanced Accuracy : 0.7279
##
##        'Positive' Class : 0
##
```

As you can see if we take 0.8 as the threshold we get an accuracy of 0.89 and a sensibility of 0.9. We will continue deleting the variable resident type because it's not hace any significance in the model.

```r
class_weight <- ifelse(train$stroke == 1, 25, 1.04)
log_reg_model2 = glm(stroke~(gender + age + hypertension + heart_disease + ever_married +work_type + av
summary(log_reg_model2)
```

```
##
## Call:
## glm(formula = stroke ~ (gender + age + hypertension + heart_disease +
##     ever_married + work_type + avg_glucose_level + bmi + smoking_status),
##     family = binomial(link = "logit"), data = train, weights = class_weight)
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -3.845e+00  2.474e-01 -15.543  < 2e-16 ***
## genderMale                 -5.104e-02  6.004e-02  -0.850 0.395203
## age                         7.919e-02  2.319e-03  34.147  < 2e-16 ***
## hypertension1               8.483e-01  7.758e-02  10.935  < 2e-16 ***
## heart_disease1              3.795e-01  1.004e-01   3.781 0.000156 ***
## ever_marriedYes            -1.188e-01  9.254e-02  -1.284 0.199235
## work_typeGovt_job          -1.355e+00  2.610e-01  -5.192 2.08e-07 ***
```

```
## work_typeNever_worked      -1.122e+01  1.329e+02  -0.084 0.932721
## work_typePrivate           -1.288e+00  2.548e-01  -5.057 4.26e-07 ***
## work_typeSelf-employed     -1.549e+00  2.679e-01  -5.784 7.29e-09 ***
## avg_glucose_level           3.103e-03  5.849e-04   5.306 1.12e-07 ***
## bmi                         1.609e-02  4.509e-03   3.569 0.000359 ***
## smoking_statusnever smoked -3.382e-01  7.673e-02  -4.408 1.05e-05 ***
## smoking_statussmokes        2.259e-01  8.853e-02   2.552 0.010720 *
## smoking_statusUnknown      -5.592e-01  9.272e-02  -6.031 1.63e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11203.1  on 3928  degrees of freedom
## Residual deviance:  7500.7  on 3914  degrees of freedom
## AIC: 7375.8
##
## Number of Fisher Scoring iterations: 12
```

```
fitted.results <- predict(log_reg_model2,newdata = test,type='response')
fitted.results <- ifelse(fitted.results > 0.8,1,0)
test$accu = fitted.results
misClasificError <- mean(fitted.results != test$stroke)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.892747701736466"
```

```
confusionMatrix(as.factor(test$accu),test$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 851  19
##          1  86  23
##
##                Accuracy : 0.8927
##                  95% CI : (0.8717, 0.9114)
##     No Information Rate : 0.9571
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2587
##
##  Mcnemar's Test P-Value : 1.187e-10
##
##             Sensitivity : 0.9082
##             Specificity : 0.5476
##          Pos Pred Value : 0.9782
##          Neg Pred Value : 0.2110
##              Prevalence : 0.9571
##          Detection Rate : 0.8693
##    Detection Prevalence : 0.8887
##       Balanced Accuracy : 0.7279
```

```
##
##        'Positive' Class : 0
##
```

The AIC was reduce so the model improved just a little.

```
anova(log_reg_model2,log_reg_model)
```

```
## Analysis of Deviance Table
##
## Model 1: stroke ~ (gender + age + hypertension + heart_disease + ever_married +
##     work_type + avg_glucose_level + bmi + smoking_status)
## Model 2: stroke ~ (gender + age + hypertension + heart_disease + ever_married +
##     work_type + Residence_type + avg_glucose_level + bmi + smoking_status)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3914     7500.7
## 2      3913     7500.3  1  0.38543   0.5347
```

As the *p-value* is higher than 0.05 we cannot reject that the simpler model is better. So we are going to deleted the variable gender because i doesn't affect the model.

```
class_weight <- ifelse(train$stroke == 1, 25, 1.04)
log_reg_model3 = glm(stroke~(ever_married + age + hypertension + heart_disease+work_type + avg_glucose_l
summary(log_reg_model3)
```

```
##
## Call:
## glm(formula = stroke ~ (ever_married + age + hypertension + heart_disease +
##     work_type + avg_glucose_level + bmi + smoking_status), family = binomial(link = "logit"),
##     data = train, weights = class_weight)
##
## Coefficients:
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -3.871e+00  2.456e-01 -15.761  < 2e-16 ***
## ever_marriedYes              -1.170e-01  9.243e-02  -1.266 0.205554
## age                           7.913e-02  2.318e-03  34.146  < 2e-16 ***
## hypertension1                 8.493e-01  7.758e-02  10.947  < 2e-16 ***
## heart_disease1                3.682e-01  9.943e-02   3.703 0.000213 ***
## work_typeGovt_job            -1.347e+00  2.608e-01  -5.163 2.43e-07 ***
## work_typeNever_worked        -1.122e+01  1.330e+02  -0.084 0.932782
## work_typePrivate             -1.283e+00  2.547e-01  -5.038 4.71e-07 ***
## work_typeSelf-employed       -1.542e+00  2.677e-01  -5.759 8.48e-09 ***
## avg_glucose_level             3.073e-03  5.837e-04   5.264 1.41e-07 ***
## bmi                           1.612e-02  4.508e-03   3.576 0.000349 ***
## smoking_statusnever smoked   -3.298e-01  7.609e-02  -4.334 1.46e-05 ***
## smoking_statussmokes          2.259e-01  8.857e-02   2.550 0.010762 *
## smoking_statusUnknown        -5.577e-01  9.267e-02  -6.019 1.76e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 11203.1  on 3928  degrees of freedom
## Residual deviance:  7501.4  on 3915  degrees of freedom
## AIC: 7374.5
##
## Number of Fisher Scoring iterations: 12
```

```
fitted.results <- predict(log_reg_model3,newdata = test,type='response')
fitted.results <- ifelse(fitted.results > 0.8,1,0)
test$accu = fitted.results
misClasificError <- mean(fitted.results != test$stroke)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.889683350357508"
```

```
confusionMatrix(as.factor(test$accu),test$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 848  19
##          1  89  23
##
##                Accuracy : 0.8897
##                  95% CI : (0.8684, 0.9086)
##     No Information Rate : 0.9571
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.252
##
##  Mcnemar's Test P-Value : 3.147e-11
##
##             Sensitivity : 0.9050
##             Specificity : 0.5476
##          Pos Pred Value : 0.9781
##          Neg Pred Value : 0.2054
##              Prevalence : 0.9571
##          Detection Rate : 0.8662
##    Detection Prevalence : 0.8856
##       Balanced Accuracy : 0.7263
##
##        'Positive' Class : 0
##
```

```
anova(log_reg_model3,log_reg_model2)
```

```
## Analysis of Deviance Table
##
## Model 1: stroke ~ (ever_married + age + hypertension + heart_disease +
##     work_type + avg_glucose_level + bmi + smoking_status)
## Model 2: stroke ~ (gender + age + hypertension + heart_disease + ever_married +
##     work_type + avg_glucose_level + bmi + smoking_status)
```

```
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3915     7501.4
## 2      3914     7500.7  1  0.72316   0.3951
```

As the *p-value* is higher than 0.05 we cannot reject that the simpler model is better. So we are going to deleted the variable ever married because i doesn't affect the model.

```
class_weight <- ifelse(train$stroke == 1, 25, 1.04)
log_reg_model4 = glm(stroke~(age + hypertension + heart_disease+work_type + avg_glucose_level + bmi + sm
summary(log_reg_model4)
```

```
##
## Call:
## glm(formula = stroke ~ (age + hypertension + heart_disease +
##     work_type + avg_glucose_level + bmi + smoking_status), family = binomial(link = "logit"),
##     data = train, weights = class_weight)
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -3.861e+00  2.455e-01 -15.729  < 2e-16 ***
## age                          7.861e-02  2.284e-03  34.412  < 2e-16 ***
## hypertension1                8.531e-01  7.740e-02  11.023  < 2e-16 ***
## heart_disease1               3.719e-01  9.928e-02   3.746 0.000180 ***
## work_typeGovt_job           -1.417e+00  2.556e-01  -5.544 2.96e-08 ***
## work_typeNever_worked       -1.120e+01  1.323e+02  -0.085 0.932489
## work_typePrivate            -1.355e+00  2.490e-01  -5.443 5.24e-08 ***
## work_typeSelf-employed      -1.615e+00  2.621e-01  -6.165 7.07e-10 ***
## avg_glucose_level            3.020e-03  5.822e-04   5.186 2.14e-07 ***
## bmi                          1.580e-02  4.505e-03   3.507 0.000454 ***
## smoking_statusnever smoked -3.190e-01  7.559e-02  -4.220 2.44e-05 ***
## smoking_statussmokes         2.288e-01  8.858e-02   2.583 0.009806 **
## smoking_statusUnknown       -5.505e-01  9.252e-02  -5.950 2.68e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 11203  on 3928  degrees of freedom
## Residual deviance:  7503  on 3916  degrees of freedom
## AIC: 7374.1
##
## Number of Fisher Scoring iterations: 12
```

```
fitted.results <- predict(log_reg_model4,newdata = test,type='response')
fitted.results <- ifelse(fitted.results > 0.8,1,0)
test$accu = fitted.results
misClasificError <- mean(fitted.results != test$stroke)
print(paste('Accuracy',1-misClasificError))
```

```
## [1] "Accuracy 0.888661899897855"
```

```r
confusionMatrix(as.factor(test$accu),test$stroke)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 848  20
##          1  89  22
##
##                Accuracy : 0.8887
##                  95% CI : (0.8673, 0.9077)
##     No Information Rate : 0.9571
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2403
##
##  Mcnemar's Test P-Value : 7.356e-11
##
##             Sensitivity : 0.9050
##             Specificity : 0.5238
##          Pos Pred Value : 0.9770
##          Neg Pred Value : 0.1982
##              Prevalence : 0.9571
##          Detection Rate : 0.8662
##    Detection Prevalence : 0.8866
##       Balanced Accuracy : 0.7144
##
##        'Positive' Class : 0
##
```

```r
anova(log_reg_model,log_reg_model4)
```

```
## Analysis of Deviance Table
##
## Model 1: stroke ~ (gender + age + hypertension + heart_disease + ever_married +
##     work_type + Residence_type + avg_glucose_level + bmi + smoking_status)
## Model 2: stroke ~ (age + hypertension + heart_disease + work_type + avg_glucose_level +
##     bmi + smoking_status)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3913     7500.3
## 2      3916     7503.0 -3  -2.7142   0.4378
```

The simpler model is better so as all the variables are significance in the model we will leave it at that.

```r
exp(log_reg_model4$coefficients)
```

```
##                (Intercept)                        age
##               0.0210381608               1.0817842993
##              hypertension1              heart_disease1
##               2.3469982988               1.4504478164
##          work_typeGovt_job      work_typeNever_worked
```

```
##               0.2425085020               0.0000136226
##           work_typePrivate     work_typeSelf-employed
##               0.2578932255               0.1988067596
##          avg_glucose_level                        bmi
##               1.0030242014               1.0159224576
## smoking_statusnever smoked       smoking_statussmokes
##               0.7268595064               1.2570488997
##      smoking_statusUnknown
##               0.5766712974
```

With the exp of the coefficients we get the odds of getting a stroke. Like the intercept means that if all the other variables are zero you have a 2.1% odds of getting a stroke. As we said in the preliminary analysis if you got hypertension the odd of getting a stroke duplicate. If you got a heart disease you have a 4.5% increase in your odds.

VSM (soroush)

XGBoost for classification (soroush)

H2O models with autoh2o (Flo)

Split the data to get a 80% for training and 20% for testing.#flo

```
h2o.init()
```

```
##  Connection successful!
##
## R is connected to the H2O cluster:
##     H2O cluster uptime:         10 hours 18 minutes
##     H2O cluster timezone:       Europe/Paris
##     H2O data parsing timezone:  UTC
##     H2O cluster version:        3.44.0.3
##     H2O cluster version age:    10 months and 4 days
##     H2O cluster name:           H2O_started_from_R_flore_dfs870
##     H2O cluster total nodes:    1
##     H2O cluster total memory:   3.20 GB
##     H2O cluster total cores:    16
##     H2O cluster allowed cores:  16
##     H2O cluster healthy:        TRUE
##     H2O Connection ip:          localhost
##     H2O Connection port:        54321
##     H2O Connection proxy:       NA
##     H2O Internal Security:      FALSE
##     R Version:                  R version 4.4.1 (2024-06-14 ucrt)
```

```
stroke_h2o=as.h2o(stroke_data)
```

```
##   |                                                                              |
```

```
split_data = h2o.splitFrame(data=stroke_h2o,ratios=0.8,seed=23)
train = split_data[[1]]
test = split_data[[2]]
pred = c("gender","age","hypertension","heart_disease","ever_married","work_type","Residence_type","avg
aml = h2o.automl(x=pred,y="stroke",training_frame=train,max_models=10,seed=23)
```

```
##   |                                                                              |
## 19:59:15.987: AutoML: XGBoost is not available; skipping it.  |
```

```
lb <- aml@leaderboard
print(lb, n = nrow(lb))
```

```
##                                                  model_id       auc   logloss
## 1  StackedEnsemble_BestOfFamily_1_AutoML_5_20241024_195915 0.8396779 0.1600156
## 2     StackedEnsemble_AllModels_1_AutoML_5_20241024_195915 0.8372433 0.1606547
## 3                             GBM_1_AutoML_5_20241024_195915 0.8346286 0.1614168
## 4                             GLM_1_AutoML_5_20241024_195915 0.8290003 0.1624284
## 5                             XRT_1_AutoML_5_20241024_195915 0.8288891 0.1630358
## 6                             GBM_2_AutoML_5_20241024_195915 0.8274151 0.1678570
## 7                             GBM_5_AutoML_5_20241024_195915 0.8177637 0.1735302
## 8                             GBM_3_AutoML_5_20241024_195915 0.8171340 0.1718411
## 9                             GBM_4_AutoML_5_20241024_195915 0.8133112 0.1755158
## 10              GBM_grid_1_AutoML_5_20241024_195915_model_1 0.7883794 0.1821893
## 11                            DRF_1_AutoML_5_20241024_195915 0.7808002 0.2537235
## 12                  DeepLearning_1_AutoML_5_20241024_195915 0.7707174 0.2066694
##        aucpr mean_per_class_error      rmse        mse
## 1  0.1671444            0.2433580 0.2083760 0.04342058
## 2  0.1673876            0.2674561 0.2085578 0.04349634
## 3  0.1706276            0.2607080 0.2086072 0.04351696
## 4  0.1682979            0.2860339 0.2088325 0.04361102
## 5  0.1704705            0.2832872 0.2094405 0.04386534
## 6  0.1666609            0.3089510 0.2124627 0.04514039
## 7  0.1622074            0.3201320 0.2158315 0.04658323
## 8  0.1542809            0.2865662 0.2140796 0.04583009
## 9  0.1435664            0.3228786 0.2165414 0.04689020
## 10 0.1473092            0.3400733 0.2158512 0.04659176
## 11 0.1267397            0.2850789 0.2188276 0.04788552
## 12 0.1241190            0.3484805 0.2190206 0.04797004
##
## [12 rows x 7 columns]
```